

# Forcing upper $\Sigma$ -uniformization in the presence of lower $\Pi$ -reduction or uniformization

Stefan Hoffelner\*

November 7, 2025

## Abstract

We present a method which allows the combination of forcing uniformization on the  $\Pi$ - and the  $\Sigma$ -side of the projective hierarchy to a certain extent. Using this method we construct a universe where  $\Pi_3^1$ -reduction holds,  $\Pi_3^1$ -uniformization fails, yet  $\Sigma_n^1$  uniformization is true for  $n \geq 4$ . We also construct a universe where  $\Pi_3^1$ -uniformization holds and for every  $n \geq 4$ ,  $\Sigma_4^1$ -uniformization holds, lowering best known upper bound for this statement from the existence of two Woodin cardinals to  $Con(\text{ZFC})$ .

## 1 Introduction

The problem of finding well-behaved choice functions for families of sets, a core concern in descriptive set theory, is a long-studied subject. Specifically, the uniformization problem, first posed by N. Lusin in 1930, asks whether one can find a choice function (called a uniformization) that has the same complexity (i.e. sits on the same projective level) as the set it is defined from.

**Definition 1.1.** *For a set  $A \subset 2^\omega \times 2^\omega$  (a set of pairs of reals), a partial function  $f$  is a uniformization of  $A$  if  $f$  picks exactly one  $y$  for each  $x$  such that there is a  $y'$  with  $(x, y') \in A$ . In other words, the graph of  $f$  is a subset of  $A$ , and the domain of  $f$  is the projection of  $A$  onto its first coordinate. A projective pointclass  $\Gamma \in \{\Sigma_n^1 \mid n \in \omega\} \cup \{\Pi_n^1 \mid n \in \omega\}$  has the uniformization property if every set in  $\Gamma$  can be uniformized by a function whose graph also belongs to  $\Gamma$ .*

Closely related to the uniformization property is a second classical regularity property, the reduction property, introduced in 1936 by K. Kuratowski.

---

\*This research was funded in whole by the Austrian Science Fund (FWF) Grant-DOI 10.55776/P37228.

**Definition 1.2.** We say that a projective pointclass  $\Gamma \in \{\Sigma_n^1 \mid n \in \omega\} \cup \{\Pi_n^1 \mid n \in \omega\}$  satisfies the  $\Gamma$ -reduction property (or just reduction) if every pair  $B_0, B_1$  of  $\Gamma$ -subsets of the reals can be reduced by a pair of  $\Gamma$ -sets  $R_0, R_1$ , which means that  $R_0 \subset B_0$ ,  $R_1 \subset B_1$ ,  $R_0 \cap R_1 = \emptyset$  and  $R_0 \cup R_1 = B_0 \cup B_1$ .

It follows immediately from the definitions that  $\Gamma$ -uniformization implies  $\Gamma$ -reduction.

It is a classical result by M. Kondo that the family of  $\Pi_1^1$  sets possesses the uniformization property, which also implies the property for  $\Sigma_2^1$  sets. Standard set theory (ZFC) cannot prove uniformization for any pointclass of higher complexity.

To prove uniformization for more complex sets, one must appeal to additional set-theoretic axioms. In the constructible universe  $L$ , for every  $n \geq 2$ ,  $\Sigma_n^1$  does have the uniformization property which follows from the existence of a good  $\Delta_2^1$ -wellorder by an old result of Addison (see [1]). Recall that a  $\Delta_n^1$ -definable wellorder  $<$  of the reals is a *good*  $\Delta_n^1$ -wellorder if  $<$  is of ordertype  $\omega_1$  and the relation  $<_I \subset (\omega^\omega)^2$  defined via

$$x <_I y \Leftrightarrow \{(x)_n : n \in \omega\} = \{z : z < y\}$$

where  $(x)_n$  is some fixed recursive partition of  $x$  into  $\omega$ -many reals, is a  $\Delta_n^1$ -definable relation. Addison's observation is that a good  $\Delta_n^1$ -wellorder implies that the  $\Sigma_m^1$ -uniformization property is true for every  $m \geq n$ . It is easy to check that the canonical wellorder of the reals in  $L$  is a good  $\Delta_2^1$ -wellorder so the  $\Sigma_n^1$ -uniformization property follows for  $n \geq 2$ .

On the other hand, and more importantly, if we assume the existence of large cardinals or alternatively strong forcing axioms, we obtain a very different picture. Due to Moschovakis (see [17] or [14], Theorem 39.9), the axiom of projective determinacy (PD) implies that  $\Pi_{2n+1}^1$  and  $\Sigma_{2n+2}^1$ -sets have the uniformization property. By the Martin-Steel theorem (see [15] or [18], Theorem 13.6.), the assumption of infinitely many Woodin cardinals implies PD, and hence large cardinals fully settle the behaviour of the uniformization property within the projective hierarchy.

An old result of Novikov (see [18], Lemma 7.25) states that  $\Gamma$ -reduction rules out reduction for the projective pointclass dual to  $\Gamma$ , denoted by  $\check{\Gamma}$  so in particular the two possibilities for the behaviour of uniformization (and hence of reduction) within the projective hierarchy contradict each other.

There are set theoretic universes which combine the behaviour of uniformization under  $V = L$  and under PD to some extent. Paradigmatic example are the canonical inner models with  $n$ -many Woodin cardinals  $M_n$ . Due to Steel, these models have good  $\Sigma_{n+2}^1$ -definable well-orders of its reals so in particular  $\Sigma_{m+2}^1$  uniformization holds there for  $m \geq n$ . Yet the presence of the  $n$ -many Woodin cardinals implies  $\Pi_n^1$ -determinacy, so in particular  $\Pi_{2m+1}^1, 2m+1 \leq n+1$ -uniformization holds for  $n$  even and  $\Pi_{2m+1}^1$ -uniformization holds for  $2m < n+1$  for  $n$  odd.

This work is a first attempt to find possibilities to combine the  $\Sigma$ -uniformization property and the  $\Pi$ -uniformization property, but using forcing techniques instead of (local forms of) projective determinacy. The articles [5] [9] and [7] deal with forcing  $\Pi_n^1$ -reduction,  $\Pi_n^1$ -uniformization for a given, fixed  $n \in \omega$ , and forcing global  $\Sigma_m^1$ -uniformization for  $m \geq 2$  respectively. The techniques introduced there are however impossible to combine. Indeed, the methods are such that they exclude themselves from each other. Forcing uniformization (or reduction) on the  $\Pi$ -side of the projective hierarchy currently needs the solution of an associated fixed point problem for sets of forcings first. These fixed point problems can only be solved if we work with iterated coding forcings which are closed under taking products. Forcing the uniformization property on the  $\Sigma$ -side according to [7], in stark contrast, needs the iteration of coding forcings which can not be closed under taking products by design (see the footnote 4 on pp. 19).

Hence it becomes a challenging problem to investigate ways to combine forcing constructions for the  $\Pi$  and the  $\Sigma$ -uniformization property. Questions of this type do have a deeper underpinning. Indeed, they can be seen as first test questions for finding constructions which would yield alternative, global, or at least less local patterns for the behaviour of the uniformization property within the projective hierarchy. The latter question is still wide open <sup>1</sup> in general and the current status of available methodology suggests that it is a very hard one. This article adds further difficulties to the picture, as its methods have a degree of limited flexibility and any attempt to force, say,  $\Pi_5^1$ -uniformization without large cardinals must be such that it can not be combined with these methods.

Main goal of this work is the introduction of a new technique which allows the combination of techniques forcing the  $\Pi$ -uniformization and methods which force the  $\Sigma$ -uniformization to some extent. This technique allows for a deep investigation of reduction and uniformization as displayed in the following first main theorem. Notably we do not assume anything beyond the consistency of ZFC for its proof.

**Theorem 1.3.** *There is a generic extension of  $L$  which satisfies*

1.  $\Pi_3^1$ -reduction,
2. a failure of  $\Pi_3^1$ -uniformization,
3.  $\Sigma_n^1$ -uniformization for every  $n \geq 4$  simultaneously.

The second main theorem is another application of ideas involved in proving the first main theorem and show that the two Woodin cardinals of  $M_2$  are not necessary to obtain the behaviour of uniformization there, instead one can get by with just assuming  $\text{Con}(\text{ZFC})$ .

---

<sup>1</sup>E.g. the problem of finding a universe where  $\Pi_3^1$  and  $\Pi_6^1$ -uniformization holds is wide open and seems to be far out of reach of the current methods.

**Theorem 1.4.** *Assume the consistency of ZFC. Then there is a universe where the  $\Pi_3^1$ -uniformization property is true and the  $\Sigma_n^1$ -uniformization property is true for all  $n \geq 4$  simultaneously.*

Due to the specific choice of the coding forcings we use, both theorems can be lifted to the  $M_n$ 's, the canonical inner models with  $n$ -many Woodin cardinals.

**Theorem 1.5.** *Let  $M_n$  denote the canonical inner model with  $n$ -many Woodin cardinals. There is a set-generic extension  $M_n[g]$  of  $M_n$  which preserves the Woodin cardinals and where additionally*

1.  $\Pi_{n+3}^1$ -reduction holds,
2.  $\Pi_{n+3}^1$ -uniformization fails,
3.  $\Sigma_{m+4}^1$ -uniformization holds for every  $m \geq n$ .

and

**Theorem 1.6.** *There is a set-generic extension  $M_n[g]$  of  $M_n$  preserving the Woodin cardinals and where additionally*

1.  $\Pi_{n+3}^1$ -uniformization holds,
2.  $\Sigma_{m+4}^1$ -uniformization holds for every  $m \geq n$ .

Put in a wider context, this work contributes to the project of a detailed investigation of separation, reduction and uniformization in the absence of large cardinals. There is a growing body of work which explores its connections, in various set theoretic contexts (see e.g. [11], [6], [8]).

We end the introduction with a short description of the organization of this article. The first four sections are devoted to re-introduce the coding machinery we use throughout which is basically the same as in [5] and [9]. The notion of allowable forcing is introduced, which is a generalization of the according notion from [5]. In the fifth section a carefully defined, generalized version of the thinning out process is presented which will be used to solve a specific fixed point problem on sets of forcings. This solution is employed to force the  $\Pi_3^1$ -reduction property, or, for the second theorem, is employed to force  $\Pi_3^1$ -uniformization. At the same time the generalized thinning out process leaves some additional room to work towards  $\Sigma_4^1$ -uniformization and a good  $\Sigma_5^1$ -wellorder of the reals. The implementation of the failure of  $\Pi_3^1$ -uniformization is tricky and involves the introduction of a  $\Sigma_3^1$ -definable  $\omega$ -sequence of reals which serve as markers which indicate potentially dangerous codes we need to ignore in order to have  $\Pi_3^1$ -reduction in the final universe. The proof of the first main theorem takes up the sixth section which is separated in several subsections to keep the presentation tidy. The proof of the second main theorem is an easier application of the ideas and methods of the proof of the first theorem and is sketched in the seventh section.

## 2 Preliminaries

The techniques of this work fully rely on [5], [9] and [11]. We shall give a short presentation of these now. We will indicate where the reader can find the proofs in detail. The coding method of our choice utilizes Suslin trees, which can be generically destroyed in an independent way of each other (see [3]).

**Definition 2.1.** Let  $\vec{T} = (T_\alpha : \alpha < \kappa)$  be a sequence of Suslin trees. We say that the sequence is an independent family of Suslin trees if for every finite set of pairwise distinct indices  $e = \{e_0, e_1, \dots, e_n\} \subset \kappa$  the product  $T_{e_0} \times T_{e_1} \times \dots \times T_{e_n}$  is a Suslin tree again.

Note that an independent sequence of Suslin trees  $\vec{T} = (T_\alpha : \alpha < \kappa)$  has the property that if  $A \subset \kappa$  and we form  $\prod_{i \in A} T_i$  with finite support, where each  $T_i$  denotes the forcing we obtain if we force with the nodes of the tree as conditions using the tree order as the partial order, then in the resulting generic extension  $V[G]$ , for every  $\alpha \notin A$ ,  $V[G] \models "T_\alpha \text{ is a Suslin tree}"$ .

**Theorem 2.2.** (see [11]) Assume that  $\aleph_1 = \aleph_1^L$  and  $M$  is an uncountable, transitive model of  $\mathbf{ZF}^- + \text{"}\aleph_1 \text{ exists"}$ . Then there is an independent sequence  $\vec{S} = (S_\alpha \mid \alpha < \omega_1)$  of Suslin trees in  $L$  and the sequence  $\vec{S}$  is uniformly  $\Sigma_1(\{\omega_1\})$ -definable over  $M$ . To be more precise, there is a  $\Sigma_1$ -formula  $\phi$  with  $\omega_1$  as the unique parameter, which does not depend on the model  $M$ , such that the relation  $\{(t, \gamma) \mid \gamma < \omega_1 \wedge t \in T^\gamma\}$ , where  $T^\gamma$  denotes the  $\gamma$ -th level of  $T$ , is definable over  $M$  using  $\phi$ .

The trees from  $\vec{S}$  will later be used to define several related coding forcings that are the main tool in our proofs.

We briefly introduce the almost disjoint coding forcing due to R. Jensen and R. Solovay ([13]). We will identify subsets of  $\omega$  with their characteristic function and will use the word reals for elements of  $2^\omega$  and subsets of  $\omega$  respectively. Let  $D = \{d_\alpha : \alpha < \aleph_1\}$  be a family of almost disjoint subsets of  $\omega$ , i.e. a family such that if  $r, s \in D$  then  $r \cap s$  is finite. Let  $X \subset \omega$  be a set of ordinals. Then there is a ccc forcing, the almost disjoint coding  $\mathbb{A}_D(X)$  which adds a new real  $x$  which codes  $X$  relative to the family  $D$  in the following way

$$\alpha \in X \text{ if and only if } x \cap d_\alpha \text{ is finite.}$$

**Definition 2.3.** The almost disjoint coding  $\mathbb{A}_D(X)$  relative to an almost disjoint family  $D$  consists of conditions  $(r, R) \in [\omega]^{<\omega} \times D^{<\omega}$  and  $(s, S) < (r, R)$  holds if and only if

1.  $r \subset s$  and  $R \subset S$ .

2. If  $\alpha \in X$  and  $d_\alpha \in R$  then  $r \cap d_\alpha = s \cap d_\alpha$ .

We shall briefly discuss the  $L$ -definable,  $\aleph_1^L$ -sized almost disjoint family of reals  $D$  we will use throughout this article. The family  $D$  is the canonical almost disjoint family one obtains when recursively adding the  $<_L$ -least real  $x_\beta$  not yet chosen and replace it with  $d_\beta \subset \omega$  where this  $d_\beta$  is the real which codes the initial segments of  $x_\beta$  using some recursive bijections between  $\omega$  and  $\omega^{<\omega}$ . The definition of  $D$  is uniform over any uncountable, transitive  $\mathbf{ZF}^-$ -models  $M$  with, as we can correctly compute  $L$  up to  $\aleph_1^L$  inside  $M$  and then apply the above definition inside  $L$ 's version of  $M$ . Even more is true, if  $M$  is a countable, transitive model of  $\mathbf{ZF}^- + \text{“}\aleph_1 \text{ exists and } \aleph_1 = \aleph_1^L\text{”}$ , then  $M$  will compute  $D \upharpoonright \omega_1^M$  in a correct way. The reason is again, that  $M$  can define an initial segment of  $L$  correctly which suffices to calculate  $D \upharpoonright \omega_1^M$ .

### 3 Coding machinery

We continue with the construction of the appropriate notions of forcing which we want to use in our proof. The goal is to first define a coding forcings  $\text{Code}(x)$  for reals  $x$ , which will force for  $x$  that a certain  $\Sigma_3^1$ -formula  $\Phi(x)$  becomes true in the resulting generic extension. The coding method is basically the same as in [5] and [9] and literally the same as in [11].

In a first step we add  $\omega_1$ -many  $\omega_1$ -Cohen subsets with a countably supported product,

$$\mathbb{P}^1 := \prod_{\alpha < \omega_1} \mathbb{C}(\omega_1).$$

Note that this forcing is itself  $\sigma$ -closed so no reals are added and  $\vec{S}$  is still an independent sequence of Suslin trees. In a second step we force over  $L^{\mathbb{P}^1}$  to destroy all members of  $\vec{S}$  via generically adding an  $\omega_1$ -branch, that is we form  $\mathbb{P}^0 := \prod_{\alpha \in \omega_1} S_\alpha$  with finite support. Note that this is an  $\aleph_1$ -sized, ccc forcing over  $L$  and also  $L^{\mathbb{P}^1}$ , and the forcing is independent of the actual model in which it is computed. The two step iteration can be thus conceived as a product of two factors  $(\prod_{i < \omega_1} \mathbb{C}(\omega_1))^L \times \prod_{\alpha \in \omega_1} S_\alpha$ . In the generic extension  $\aleph_1$  is preserved and  $\text{CH}$  remains to be true.

We use  $W$  to denote this generic extension of  $L$ , that is we let  $g^0 \times g^1$  be a  $\mathbb{P}^0 \times \mathbb{P}^1$  generic filter over  $L$  where  $\mathbb{P}^0$  adds cofinal branches to each member of  $\vec{S}$  and  $\mathbb{P}^1$  adds  $\aleph_1$ -many  $\omega_1$ -Cohen subsets, then

$$W = L[g^0 \times g^1].$$

Let  $x \in W$  be a real, and let  $m, k \in \omega$  and let  $\eta < \omega_1$ . We simply write  $(x, m, k, 1)$  for a real  $w$  which codes the quadruple  $(x, m, k, 1)$  in a recursive way. The forcing  $\text{Code}(x, m, k, 1, \eta)$ <sup>2</sup> which codes the quadruple  $(x, m, k, 1)$

<sup>2</sup>The other coding forcing,  $\text{Code}(x, m, k, 0, \eta)$ , which codes quadruples  $(x, m, k, 0)$  instead is defined in the analogue way

into  $\vec{S}$  is defined as the almost disjoint coding forcing of a specific set  $Y \subset \omega_1$ , that is

$$\text{Code}(m, k, 1, x, \eta) := \mathbb{A}(Y).$$

We will define the crucial set  $Y \subset \omega_1$  now.

To ease notation we let  $g \subset \omega_1$  be  $g_\eta^1$  for  $\eta < \omega_1$ , where  $g_\eta^1$  is the  $\eta$ -th coordinate of the  $\prod_{\alpha < \omega_1} \mathbb{C}(\omega_1)$ -generic filter over  $L[g^0]$ . We let  $\rho : ([\omega_1]^\omega)^L \rightarrow \omega_1$  be some canonically definable, constructible bijection between these two sets. We use  $\rho$  and  $g$  to define the set  $h \subset \omega_1$ , which eventually shall be the set of indices of  $\omega$ -blocks of  $\vec{S}$ , where we “code up the characteristic function of the real  $(m, k, 1, x)$ ”, the latter slogan will be made precise in a moment. Let

$$h := \{\rho(g \cap \alpha) : \alpha < \omega_1\}$$

and let

$$A := \{\omega\gamma + 2n \mid \gamma \in h, n \notin (m, k, 1, x)\} \cup \\ \{\omega\gamma + 2n + 1 \mid \gamma \in h, n \in (m, k, 1, x)\}.$$

Let  $X \subset \omega_1$  be chosen such that it codes the following objects:

- The set  $A \subset \omega_1$ .
- Some set  $\{b_\beta \subset S_\beta \mid \beta \in A\}$  of  $\omega_1$ -branches. We demand that for every  $\beta \in A$ ,  $b_\beta$  is a  $L[g^0]$ -generic branch for the forcing  $S_\beta \in \vec{S}$ .

Note that, when working in  $L[X]$  and if  $\gamma \in h$  then we can read off  $(m, k, 1, x)$ , and thus we say that  $(m, k, 1, x)$  is coded into  $\vec{S}$  at the  $\omega$ -block starting at  $\gamma$ , via looking at the  $\omega$ -block of  $\vec{S}$ -trees starting at  $\gamma$  and determine which tree has an  $\omega_1$ -branch in  $L[X]$ .

(\*) $_\gamma$   $n \in (m, k, 1, x)$  if and only if  $S_{\omega \cdot \gamma + 2n + 1}$  has an  $\omega_1$ -branch, and  $n \notin (m, k, 1, x)$  if and only if  $S_{\omega \cdot \gamma + 2n}$  has an  $\omega_1$ -branch.

Indeed if  $n \notin (m, k, 1, x)$  then we added a cofinal branch through  $S_{\omega \cdot \gamma + 2n}$ . If on the other hand  $S_{\omega \cdot \gamma + 2n}$  does not have an  $\omega_1$ -branch in  $L[X]$  then we must have added an  $\omega_1$ -branch through  $S_{\omega \cdot \gamma + 2n + 1}$  as we always add an  $\omega_1$ -branch through either  $S_{\omega \cdot \gamma + 2n + 1}$  or  $S_{\omega \cdot \gamma + 2n}$  and adding branches through some  $S_\alpha$ 's will not affect that some  $S_\beta$  remain Suslin in  $L[X]$ , as  $\vec{S}$  is independent.

We note that we can apply an argument resembling David's trick ([2]) in this situation. We rewrite the information of  $X \subset \omega_1$  as a subset  $Y \subset \omega_1$  using the following line of reasoning. Keeping lemma 2.2 in mind, it is clear that any transitive,  $\aleph_1$ -sized model  $M$  of  $\text{ZF}^-$  which contains  $X$  will be able to first define  $\vec{S}$  correctly and also correctly decode out of  $X$  all the information regarding  $(m, k, 1, x)$  being coded at each  $\omega$ -block of  $\vec{S}$  starting at every  $\gamma \in h$ . Consequently, if we code the model  $(M, \in)$  which contains  $X$  as a set  $X_M \subset \omega_1$ , then for any uncountable  $\beta$  such that  $L_\beta[X_M] \models \text{ZF}^-$ :

$L_\beta[X_M] \models$  “The model decoded out of  $X_M$  satisfies  $(*)_\gamma$  for every  $\gamma \in h$ ”.

In particular there will be an  $\aleph_1$ -sized ordinal  $\beta$  as above and we can fix a club  $C \subset \omega_1$  and a sequence  $(M_\alpha : \alpha \in C)$  of countable elementary submodels of  $L_\beta[X_M]$  such that

$$\forall \alpha \in C (M_\alpha < L_\beta[X_M] \wedge M_\alpha \cap \omega_1 = \alpha)$$

Now let the set  $Y \subset \omega_1$  code the pair  $(C, X_M)$  such that the odd entries of  $Y$  should code  $X_M$  and if  $E(Y)$  denotes the set of even entries of  $Y$  and  $\{c_\alpha : \alpha < \omega_1\}$  is the enumeration of  $C$  then

1.  $E(Y) \cap \omega$  codes a well-ordering of type  $c_0$ .
2.  $E(Y) \cap [\omega, c_0) = \emptyset$ .
3. For all  $\beta$ ,  $E(Y) \cap [c_\beta, c_{\beta+\omega})$  codes a well-ordering of type  $c_{\beta+1}$ .
4. For all  $\beta$ ,  $E(Y) \cap [c_\beta + \omega, c_{\beta+1}) = \emptyset$ .

We obtain

- (\*\*) For any countable transitive model  $M$  of “ $\text{ZF}^-$  and  $\aleph_1$  exists” such that  $\omega_1^M = (\omega_1^L)^M$  and  $Y \cap \omega_1^M \in M$ ,  $M$  can construct its version of the universe  $L[Y \cap \omega_1^N]$ , and the latter will see that there is an  $\aleph_1^M$ -sized transitive model  $N \in L[Y \cap \omega_1^N]$  which models  $(*)$  for  $(m, k, 1, x)$  and every  $\gamma \in h \cap M$ .

Thus we have a local version of the property  $(*)$ . We have finally defined the desired set  $Y$  and now we use

$$\text{Code}(m, k, 1, x, \eta) := \mathbb{A}(Y)$$

relative to our previously defined, almost disjoint family of reals  $D \in L$  (see the paragraph after Definition 2.5) to code the set  $Y$  into one real  $r$ . This forcing only depends on the subset of  $\omega_1$  we code, thus  $\mathbb{A}_D(Y)$  will be independent of the surrounding universe in which we define it, as long as it has the right  $\omega_1$  and contains the set  $Y$ . The effect of the coding forcing  $\text{Code}(m, k, x, 1, \eta)$  is that it generically adds a real  $r$  such that

- (\*\*\*) For any countable, transitive model  $M$  of “ $\text{ZF}^-$  and  $\aleph_1$  exists”, such that  $\omega_1^M = (\omega_1^L)^M$  and  $r \in M$ ,  $M$  can construct its version of  $L[r]$ , denoted by  $L[r]^M$ , which in turn thinks that there is a transitive  $\text{ZF}^-$ -model  $N$  of size  $\aleph_1^M$  such that  $N$  believes  $(*)$  for  $(m, k, 1, x)$  and every  $\gamma \in h \cap M$ .

Indeed, if  $r$  and  $M$  are as above, then  $M$  and  $L[r]^M$  will compute the almost disjoint family  $D$  up to the real indexed with  $\omega_1 \cap M$  correctly, as discussed below the definition 2.3. As a consequence,  $L[r]^M$  will contain the set  $Y \cap \omega_1^M$ , where  $Y \subset \omega_1$  is as in (\*\*). So in  $L[Y \cap \omega_1^M]$ , there is an  $\aleph_1^M$ -sized, transitive  $N$  which models  $(*)_\gamma$  for every  $\gamma \in h \cap M$ , as claimed.

Note that (\*\*\*) is a  $\Pi_2^1$ -formula in the parameters  $r$  and  $(m, k, 1, x)$ , as the set  $h \cap M \subset \omega_1^M$  is coded into  $r$ . We say in the above situation that the real  $(m, k, 1, x)$  is written into  $\vec{S}$ , or that  $(m, k, 1, x)$  is coded into  $\vec{S}$ . To summarize our discussion, given an arbitrary real of the form  $(m, k, 1, x)$ , then our forcing  $\text{Code}(m, k, 1, x)$ , when applied over  $W$ , will add a real  $r$  which will turn the  $\Pi_2^1$ -formula (\*\*\*) for  $r, (m, k, 1, x)$  into a true statement in  $W^{\text{Code}(m, k, 1, x)}$ .

The coding forcing which codes a given real  $(m, k, 1, x)$  into the  $\vec{S}$ , denoted by  $\text{Code}(m, k, 0, x)$  is defined in the same way. The projective and local statement (\*\*\*), if true, will determine how certain inner models of the surrounding universe will look like with respect to branches through  $\vec{S}$ . That is to say, if we assume that (\*\*\*) holds for a real  $(m, k, 1, x)$  and is the truth of it is witnessed by a real  $r$ . Then  $r$  also witnesses the truth of (\*\*\*) for any transitive model  $M$  of the theory “ $\text{ZF}^- + \aleph_1$  exists and  $\aleph_1 = \aleph_1^L$ ”, which contains  $r$  (i.e. we can drop the assumption on the countability of  $M$ ). Indeed if we assume that there would be an uncountable, transitive  $M, r \in M$ , which witnesses that (\*\*\*) is false. Then by Löwenheim-Skolem, there would be a countable  $N < M, r \in N$  which we can transitively collapse to obtain the transitive  $\bar{N}$ . But  $\bar{N}$  would witness that (\*\*\*) is not true for every countable, transitive model, which is a contradiction. Consequently, the real  $r$  carries enough information that the universe  $L[r]$  will see that certain trees from  $\vec{S}$  have branches in that

$$n \in (m, k, 1, x) \Rightarrow L[r] \models “S_{\omega_{\gamma+2n+1}} \text{ has an } \omega_1\text{-branch}”.$$

and

$$n \notin (m, k, 1, x) \Rightarrow L[r] \models “S_{\omega_{\gamma+2n}} \text{ has an } \omega_1\text{-branch}”.$$

Indeed, the universe  $L[r]$  will see that there is a transitive model  $N$  of “ $\text{ZF}^- + \aleph_1$  exists and  $\aleph_1 = \aleph_1^L$ ” which believes  $(*)$  for every  $\gamma \in h \subset \omega_1$ , the latter being coded into  $r$ . But by upwards  $\Sigma_1$ -absoluteness, and the fact that  $N$  can compute  $\vec{S}$  correctly, if  $N$  thinks that some tree in  $\vec{S}$  has a branch, then  $L[r]$  must think so as well.

## 4 Allowable forcings

Next we define the set of forcings which we will use in our proof. We aim to iterate the coding forcings we defined in the last section. We need to

generalize the notion of allowable, introduced in [5], as we want to force towards

- $\Pi_3^1$ -reduction,
- the failure of  $\Pi_3^1$ -uniformization,
- $\Sigma_4^1$ -uniformization and
- a good  $\Sigma_5^1$ -well-order.

These different tasks will be tackled using iterated coding forcings which obey the following definition.

**Definition 4.1.** *Let  $W$  be our ground model. Let  $\alpha < \omega_1$  and let  $F \in W$ ,  $F : \alpha \rightarrow W$  be a bookkeeping function. A finite support iteration  $\mathbb{P} = ((\mathbb{P}_\beta, \dot{Q}_\beta) : \beta < \alpha)$  is called allowable (relative to the bookkeeping function  $F$ ) if the function  $F : \alpha \rightarrow W$  determines  $\mathbb{P}$  inductively as follows:*

- (1) *We assume that  $\beta \geq 0$  and  $\mathbb{P}_\beta$  is defined. We let  $G_\beta$  be a  $\mathbb{P}_\beta$ -generic filter over  $W$  and assume that  $F(\beta) = (\dot{m}, \dot{k}, \dot{l}, \dot{x}, \dot{\eta})$ , for a quintuple of  $\mathbb{P}_\beta$ -names. We assume that  $\dot{x}^{G_\beta} =: x$  is a real,  $\dot{m}^{G_\beta} =: m$  and  $\dot{k}^{G_\beta}$  are natural numbers which code two  $\Pi_3^1$ -sets  $A_m$  and  $A_k$  respectively,  $\dot{l}^{G_\beta} \in \{0, 1\}$  and  $\dot{\eta}^{G_\beta}$  is an ordinal  $< \omega_1$ .*

*Then we split into two cases:*

- *If there is a  $\gamma < \beta$  and a  $\mathbb{P}_\gamma$ -name of a triple  $(\dot{m}', \dot{k}', \dot{l}', \dot{a}, \dot{\eta})$  such that  $\dot{a}^{G_\gamma} = a \in \omega^\omega$ ,  $\dot{m}'^{G_\gamma} = m' \in \omega$ ,  $\dot{k}'^{G_\gamma} = k' \in \omega$ ,  $\dot{\eta}^{G_\gamma} = \eta$  and  $F(\gamma) = (\dot{m}, \dot{k}, \dot{l}, \dot{a}, \dot{\eta})$ , then we force with the trivial forcing. We say in this situation that  $\eta$  has already been used for coding, or  $\eta$  is not free.*
- *If not, then let*

$$\dot{Q}_\beta^{G_\beta} = \mathbb{P}(\beta)^{G_\beta} := \text{Code}(m, k, l, x, \eta).$$

*We say that in this situation  $\eta$  is free or  $\eta$  has not been used for coding yet.*

- (2) *If on the other hand,  $F(\beta) = (\dot{m}, \dot{x}, \dot{y}, \dot{\eta})$  and  $\dot{m}$  is the Gödel number of a  $\Pi_3^1$ -set in the plane,  $\dot{x}, \dot{y}$  are both  $\mathbb{P}_\beta$ -names of reals,  $\dot{\eta}$  a name for a countable ordinal, then we define*

$$\dot{Q}_\beta^{G_\beta} := \text{Code}(0, 0, \dot{x}^{G_\beta}, \dot{y}^{G_\beta}, \dot{\eta}^{G_\beta})$$

*provided  $\eta$  is free. If  $\eta$  is not free we let  $\eta'$  be the least ordinal which is free and use  $\dot{Q}_\beta^{G_\beta} := \text{Code}(0, 0, \dot{x}^{G_\beta}, \dot{y}^{G_\beta}, \dot{\eta}^{G_\beta})$  instead. To avoid an*

ambiguity in the definition we also declare that 0 is not the Gödel number of any formula. The seemingly redundant information when coding  $(0, 0, x, y)$  instead of just  $(0, x, y)$  is to avoid an ambiguous definition again.

- (3) If  $F(\beta) = (\dot{m}, \dot{x}, \dot{y}, \dot{a}_0, \dot{a}_1, \dot{\eta})$  where  $\dot{m}^{G_\beta} = m \in \omega$  and  $m$  the Gödel number of a  $\Sigma_4^1$ -formula, then we use  $\text{Code}(m, x, y, a_0, a_1, \eta)$ , provided  $\eta$  is free. Otherwise we use the least  $\eta'$  which is free and code there.
- (4) If  $F(\beta) = (\dot{m}, \dot{x}, \dot{y}, \dot{\eta})$  and  $\dot{x}, \dot{y}$  are both names of reals whereas  $\dot{m}$  is the Gödel number of a  $\Sigma_5^1$ -formula, then we use  $\text{Code}(1, 1, x, y, \eta)$ , provided  $\eta$  is free and use the least  $\eta'$  which is free otherwise for coding  $(1, 1, x, y)$  there. Again to avoid ambiguity we assume that 1 is not the Gödel number of a formula and code  $(1, 1, x, y)$  instead of just  $(1, x, y)$ .
- (5) Finally if  $F(\beta)$  is of the form  $(\dot{n}, \dot{x}, \dot{\eta})$ , for  $\dot{x}$  a  $\mathbb{P}_\beta$ -name of a real,  $\dot{n}$  a  $\mathbb{P}_\beta$ -name of a natural number and  $\dot{\eta}$  the  $\mathbb{P}_\beta$ -name of an ordinal  $< \omega_1$ , then we let

$$\dot{Q}_\beta^{G_\beta} = \mathbb{P}(\beta) := \text{Code}(n, x, \eta)$$

provided  $\eta$  is free for coding. Otherwise we let  $\eta'$  be the least ordinal which is free and code at  $\eta'$  instead. This item also explain why we coded  $(0, 0, x, y)$  in item (2) and  $(1, 1, x, y)$  in item (4).

We add as a mildly clarifying remark that the forcings from (1) (2) and (5) are used to work towards a universe where  $\Pi_3^1$ -reduction holds and  $\Pi_3^1$ -uniformization fails; and the forcings from (3) and (4) are used to force  $\Sigma_4^1$ -uniformization and a good  $\Sigma_5^1$ -well-order of the reals respectively.

Allowable forcings will form the base set of an inductively defined shrinking process, thus they are also denoted by 0-allowable with respect to  $F$  to emphasize this fact. Every allowable forcing  $\mathbb{P}$  can be written over  $L$  as<sup>3</sup>  $\mathbb{P}^0 * \mathbb{P}^1 * (*_{\alpha < \delta}(\mathbb{P}(\alpha)))$ , for  $\mathbb{P}^0 = \prod_{\alpha < \omega_1} \mathbb{C}(\omega_1)$ ,  $\mathbb{P}^1 = \prod_{\alpha < \omega_1} S_\alpha$  and the factors  $\mathbb{P}(\alpha) = \mathbb{A}(Y_\alpha)$ . We note that both  $\mathbb{P}^1$  and  $*_{\alpha < \delta}(\mathbb{P}(\alpha))$  are ccc notions of forcing, and as every instance of almost disjoint coding forcing in our allowable iteration picks exactly one coordinate of the generic of  $\mathbb{P}^0$  as a coding area, we conclude that for an arbitrary allowable forcing  $\mathbb{P}$ , there is a countable ordinal  $\alpha < \omega_1$  such that  $\mathbb{P}$  only relies on the first  $\alpha$ -many  $\mathbb{C}(\omega_1)$ -many coordinates of the  $L$ -generic filter  $G \subset \prod_{\beta < \omega_1} \mathbb{C}(\omega_1)$ . Thus if we partition  $\mathbb{P}^0$  into a part with coordinates below or equal  $\alpha$  and a part with coordinates above  $\alpha$  and write  $\mathbb{P}^0 = \prod_{i \leq \alpha} \mathbb{C}(\omega_1) \times \prod_{i > \alpha} \mathbb{C}(\omega_1)$ , then we can re-arrange the allowable  $\mathbb{P}$  as

$$\mathbb{P} = \mathbb{P}^0 * \mathbb{P}^1 * (*_{\alpha < \delta} \mathbb{P}(\alpha)) = \left( \prod_{i \leq \alpha} \mathbb{C}(\omega_1) \right) * ((\mathbb{P}^1 * (*_{\alpha < \delta} \mathbb{P}(\alpha))) \times \prod_{i > \alpha} \mathbb{C}(\omega_1)).$$

<sup>3</sup>Here and later we will simply write  $*\mathbb{P}(\alpha)$  for the forcing iteration one obtains when using the  $\mathbb{P}(\alpha)$ 's as a factor. More precisely, if  $(\mathbb{P}_\alpha, \dot{Q}_\alpha \mid \alpha < \kappa)$  is the usual notation for an iteration of length  $\kappa$ , then  $*\mathbb{Q}_\alpha$  simply denotes  $\mathbb{P}_\kappa$ .

So when working over  $L^{\prod_{i < \alpha} \mathbb{C}(\omega_1)}$ , which is an  $\omega$ -distributive generic extension of  $L$ , the allowable  $\mathbb{P}$  can be written as a product of the two factors  $\prod_{i > \alpha} \mathbb{C}(\omega_1)$  (evaluated as in  $L$  or equivalently evaluated as in  $L^{\prod_{i < \alpha} \mathbb{C}(\omega_1)}$ ) and  $\mathbb{P}^1 * (*_{\alpha < \delta} \mathbb{P}(\alpha))$ . Thus by Easton's Lemma applied over  $V = L^{\prod_{i < \alpha} \mathbb{C}(\omega_1)}$  (see Lemma 15.19 from [12]), every real in  $L^{\mathbb{P}}$ , is in fact already in the forcing extension over  $L$  using the partial order  $(\prod_{i < \alpha} \mathbb{C}(\omega_1)) * ((\mathbb{P}^1 * (*_{\alpha < \delta} \mathbb{P}(\alpha))))$  as  $(\prod_{i \geq \alpha} \mathbb{C}(\omega_1))^L$  is still  $\omega$ -distributive over the universe obtained by forcing over  $L$  with the partial order  $(\prod_{i < \alpha} \mathbb{C}(\omega_1)) * ((\mathbb{P}^1 * (*_{\alpha < \delta} \mathbb{P}(\alpha))))$ . In particular every name of a real obtained with an allowable forcing can be written as a name which depends on a countable set of coding areas only.

As a second and related remark we add that any allowable forcing  $\mathbb{P} \in W$  can be defined already correctly over a proper inner model of  $W$ . Indeed, as  $\mathbb{P}$ 's definition depends on the names of reals listed by  $F$ , we see that  $\mathbb{P}$  can be defined using a countable list of names of reals for reals in  $W$ , and additionally the  $\aleph_1$ -many branches through  $L$ -Suslin from  $\vec{S}$  trees which are used to define the sets  $Y \subset \omega_1$  which then get coded using  $\mathbb{A}_D(Y)$ . As every real in  $W$  in fact belongs to some  $L^{\prod_{i < \beta < \omega_1} S_i}$ , there are always  $\aleph_1$ -many trees from  $\vec{S}$  and  $\aleph_1$ -many coding areas which are not used when defining  $\mathbb{P}$  over  $W$ . If  $\mathbb{P} \in W$  is a forcing such that there is an  $\alpha < \omega_1$  and an  $F \in W$ ,  $F : \alpha \rightarrow W$  such that  $\mathbb{P}$  is allowable with respect to  $F$ , then we often just drop the  $F$  and simply say that  $\mathbb{P} \in W$  is allowable. As mentioned already informally, every allowable forcing uniquely defines a countable set of coding areas it uses with its coding forcings.

**Definition 4.2.** *Let  $\mathbb{P} = ((\mathbb{P}_\alpha, \dot{Q}_\alpha = \mathbb{P}(\alpha)) \mid \alpha < \delta)$  be an allowable forcing. Let  $G \subset \mathbb{P}$  be a generic filter over  $W$ . Then*

$$C^G := \{\eta < \omega_1 \mid \exists \beta < \delta \exists \dot{x}, \dot{m}, \dot{k}, \dot{l}, \dot{\eta} \in W^{\mathbb{P}_\beta} \\ \dot{Q}_\beta^{G_\beta} = \text{Code}(\dot{x}^{G_\beta}, \dot{m}^{G_\beta}, \dot{k}^{G_\beta}, \dot{l}^{G_\beta}, \dot{\eta}^{G_\beta} = \eta)\}$$

is the set of coding areas of  $\mathbb{P}$  relative to  $G$ . We also let

$$C^{\mathbb{P}} := \{\eta < \omega_1 \mid \exists p \in \mathbb{P}(p \Vdash \eta \in C^G)\}.$$

As noted already above,  $C^G$  and also  $C^{\mathbb{P}}$  will always be countable sets for every allowable  $\mathbb{P}$ . We derive some very easy properties of allowable forcings.

**Lemma 4.3.** *(see [11])*

1. *If  $\mathbb{P} = ((\mathbb{P}(\beta), \dot{Q}_\beta) : \beta < \delta) \in W$  is allowable then for every  $\beta < \delta$ ,  $\mathbb{P}_\beta \Vdash |\dot{Q}_\beta| = \aleph_1$ , thus every factor of  $\mathbb{P}$  is forced to have size  $\aleph_1$ .*
2. *Every allowable forcing over  $W$  is ccc and thus preserves cardinals.*

3. *Every allowable forcing over  $W$  preserves CH. Furthermore, if  $\mathbb{P} = (\mathbb{P}_\alpha, \dot{\mathbb{Q}}_\alpha) : \alpha < \omega_1 \in W$  is an  $\omega_1$ -length iteration such that each initial segment of the iteration is allowable over  $W$ , then  $W^{\mathbb{P}} \models \text{CH}$ .*
4. *The product of two allowable forcings  $\mathbb{P}$  and  $\mathbb{Q}$  can be densely embedded into an allowable forcing provided that  $C^{\mathbb{P}} \cap C^{\mathbb{Q}} = \emptyset$ .*

Let  $\mathbb{P} = ((\mathbb{P}_\beta, \dot{\mathbb{Q}}_\beta) : \beta < \delta)$  be an allowable forcing with respect to some  $F \in W$ . The set of (names of) reals which are enumerated by  $F$  is dubbed the set of reals which are coded by  $\mathbb{P}$ . That is, for every  $\beta$ , if we let  $\dot{x}_\beta$  be the (name) of a real listed by  $F(\beta)$  and if we let  $G \subset \mathbb{P}$  be a generic filter over  $W$  and finally if we let  $\dot{x}_\beta^G =: x_\beta$ , then we say that  $\{x_\beta : \beta < \alpha\}$  is the set of reals coded by  $\mathbb{P}$  and  $G$  (though we will suppress the  $G$ ).

A crucial property of 0-allowable forcings is that we are in full control over which reals are coded and which are not. We define

$$\Phi(x) \equiv \exists r \forall M (M \text{ is countable and transitive and } M \models \text{ZF}^- \\ \text{and } \omega_1^M = (\omega_1^L)^M \text{ and } r, x \in M \rightarrow M \models \varphi(r, x))$$

where  $\varphi(r, x)$  asserts that in  $M$ 's version of  $L[r]$ , there is a transitive,  $\aleph_1^M$ -sized model of “ZF<sup>-</sup> and  $\aleph_1$  exists” which witnesses that  $x$  is coded into  $\vec{S}$ .

We know already that for a given real  $x$ , if we force with  $\text{Code}(x, \eta)$  for some  $\eta < \omega_1$  then  $\Phi(x)$  will hold in the generic extension. There is still the possibility that an allowable forcing will add reals  $y$  which satisfy  $\Phi(y)$  without using a coding forcing of the form  $\text{Code}(y, \eta)$ . This would be a major problem as we need our coding method to be precise. The next lemma says that an allowable forcing does not accidentally add reals  $x$  which satisfy  $\Phi$ , so allowable forcings are a suitable tool for the things to come.

**Lemma 4.4.** *(see [5], [9]) If  $\mathbb{P} \in W$  is allowable,  $\mathbb{P} = ((\mathbb{P}_\beta, \dot{\mathbb{Q}}_\beta) : \beta < \delta)$ ,  $G \subset \mathbb{P}$  is generic over  $W$  and  $\{x_\beta : \beta < \delta\}$  is the set of reals which are coded by  $\mathbb{P}$ . Let  $\Phi(v_0)$  be the distinguished formula from above. Then in  $W[G]$ , the set of reals which satisfy  $\Phi(v_0)$  is exactly  $\{x_\beta : \beta < \delta\}$ .*

## 5 Thinning out

We define next a derivative of the class of allowable forcings. These derivatives can be applied transfinitely often, yielding smaller and smaller non-empty subsets of the allowable forcings. Eventually the derivative operator will act on a non-empty subset which can not be thinned out further, in other words is this subset is a fixed point under the operation. Forcings from this set will be called  $\infty$ -allowable and they are the right set of coding forcings we want to use to force  $\Pi_3^1$ -reduction. We emphasize that the other tasks, namely forcing  $\Sigma_n^1$ -uniformization for  $n \geq 4$  and the failure of

$\Pi_3^1$ -uniformization play no role in the definition of the thinning out process. These tasks will be build in later, once we are finished in our definition of the thinning out operator.

As there will be several longer definitions, we want to motivate the thinning out process now, providing some intuitions which fuel the constructions. A more detailed discussion can be found in [5].

### 5.1 Informal discussion of the idea

We proceed with an informal discussion of the main ideas of the proof. We consider an arbitrary pair  $A_m$  and  $A_k$  of  $\Pi_3^1$ -sets and want to find reducing sets  $D_{m,k}^0, D_{m,k}^1$  i.e. sets with the following properties

1.  $D_{m,k}^0, D_{m,k}^1$  are both  $\Pi_3^1$ ,
2.  $D_{m,k}^0 \subset A_m$  and  $D_{m,k}^1 \subset A_k$ ,
3.  $D_{m,k}^0 \cap D_{m,k}^1 = \emptyset$ ,
4.  $D_{m,k}^0 \cup D_{m,k}^1 = A_m \cup A_k$ .

The ansatz is to use the two types of coding forcings  $\text{Code}(m, k, 0, x, \eta)$  and  $\text{Code}(m, k, 1, x, \eta)$  to define reducing sets. for the fixed pair of  $\Pi_3^1$ -formulas,  $\varphi_m, \varphi_k$ . The two candidates for reducing sets are defined by the  $\Pi_3^1$ -formulas

$$D_{m,k}^0 := \{x : (m, k, x, 0) \text{ is not coded into } \vec{S}\},$$

and

$$D_{m,k}^1 := \{x : (m, k, x, 1) \text{ is not coded into } \vec{S}\}.$$

As always there will be a bookkeeping function  $F$  which hands us at every stage  $\beta < \omega_1$  (names of) reals  $x$  and the task is to decide which one of the two forcings,  $\text{Code}(m, k, 0, x, \eta)$  or  $\text{Code}(m, k, 1, x, \eta)$  we want to use at that stage. In other words we decide at stage  $\beta$  whether we place  $x$  into  $D_{m,k}^0$  or  $D_{m,k}^1$ .

We approach this problem as follows. As a first observation we note that if a real  $x$  is such that it can not be forced out of  $A_m$  with an allowable forcing then we can safely put into  $D_{m,k}^0$  via using the forcing  $\text{Code}(m, k, 1, x, \eta)$  for some  $\eta$  and ensure to never put  $x$  into  $D_{m,k}^1$  later. If  $x$  is such that it can be forced out of  $A_m$  with an allowable forcing but can not be forced out of  $A_k$ , then it is safe to place  $x$  into  $D_{m,k}^1$ . In the remaining case,  $x$  can be forced out of both  $A_m$  and  $A_k$ , in which situation we just let a bookkeeping function decide where to place  $x$ .

This new class of allowable forcings is a first approximation of an iteration which should solve the problem of finding reducing sets  $D_{m,k}^0$  and  $D_{m,k}^1$  and

will be called 1-allowable. We note that when forcing with a 1-allowable forcing we actually ask the wrong questions when trying to place  $x$ . Indeed if we run a 1-allowable iteration, whenever we ask whether some  $x$  can be forced out of  $A_m$  with an allowable forcing, we better should have asked whether  $x$  can be forced out of  $A_m$  with a 1-allowable forcing as this is the class our iteration belongs to.

Thus it is reasonable to add a second question at each stage of a 1-allowable forcing: given a real  $x$  we ask whether  $x$  can be forced out of  $A_m$  or  $A_k$  with a further allowable forcing and if the answer is yes, then we ask whether  $x$  can be forced out of  $A_m$  or  $A_k$  with a 1-allowable forcing. If the answer now is no, we can safely place  $x$  into  $D_{m,k}^0$  or  $D_{m,k}^1$ , as the forcing we are about to define will also be 1-allowable. This new set of forcings will be called 2-allowable. But now, again, looking at the definition of 2-allowable forcings we see that we actually ask the wrong questions again.

These considerations hint at a fixed point problem which is sitting behind the problem of making our ansatz work. The way to solve this fixed point problem is to transfinitely often repeat the above consideration which yields better and better approximations to finding the right set of iterations of coding forcings. Eventually the set of approximations will become stable at a class of forcings we call  $\infty$ -allowable. These forcings are solutions to the fixed point problem and will be employed to solve the reduction problem.

## 5.2 The derivative operator

We work with  $W$  as our ground model. Inductively we assume that for an ordinal  $\alpha$  and an arbitrary bookkeeping function  $F \in W$  mapping to  $H(\omega_2)^2$ , we have already defined the notion of  $\delta$ -allowable with respect to  $F$  for every  $\delta < \alpha$ , and the definition works uniformly for every model  $W[G]$ , where  $G$  is a generic filter for an allowable forcing. Note that these inductive requirements are met for (0-)allowable forcings. Now we aim to define the derivation of the  $< \alpha$ -allowable forcings which we call  $\alpha$ -allowable.

**Definition 5.1.** *Let  $\delta < \omega_1$  then a  $\delta$ -length iteration  $\mathbb{P}$  is called  $\alpha$ -allowable if it is recursively constructed using a bookkeeping function  $F : \delta \rightarrow H(\omega_2)^2$ , such that for every  $\beta < \delta$ ,  $F(\beta)$  is a pair  $(F(\beta)_0, F(\beta)_1)$ , and two rules at every stage  $\beta < \delta$  of the iteration. We assume inductively that we already created the forcing iteration up to  $\beta$ ,  $\mathbb{P}_\beta$  and we let  $G_\beta$  denote a hypothetical  $\mathbb{P}_\beta$ -generic filter over  $W$ . We shall now define the next forcing of our iteration  $\mathbb{P}(\beta)$ . Using the bookkeeping  $F$  we split into two cases.*

- (a) *We assume first that the first coordinate of  $F(\beta)$ ,  $(F(\beta))_0 = (\dot{x}, m, k)$ , where  $\dot{x}$  is the  $\mathbb{P}_\beta$ -name of a real and  $m < k$  are natural numbers which code two  $\Pi_3^1$ -formulas  $\varphi_m$  and  $\varphi_k$  with associated  $\Pi_3^1$ -sets  $A_m$  and  $A_k$ . Further we assume that  $\dot{x}^{G_\beta} = x$ , and  $W[G_\beta] \models x \in A_m \cup A_k$ . We assume that in  $W[G_\beta]$ , the following is true:*

There is an ordinal  $\zeta < \alpha$ , which is chosen to be minimal for which

(i) for every  $\zeta$ -allowable forcing  $\mathbb{Q} \in W[G_\beta]$  we have that, over  $W[G_\beta]$ :

$$\mathbb{Q} \Vdash x \in A_m$$

We assume that  $F(\beta)_1 = \eta < \omega_1$ . In this situation we force with  $\text{Code}(x, m, k, 1, \eta)$  provided,  $\eta$  has not been used for coding yet. Otherwise we force with  $\text{Code}(x, m, k, 1, \zeta)$ , for  $\zeta$  being the least ordinal which has not been used for coding yet.

(ii) If (i) for  $\zeta$  is false but the dual situation is true, i.e. for every  $\zeta$ -allowable forcing  $\mathbb{Q} \in W[G_\beta]$ , we have that  $W[G_\beta]$  thinks that

$$\mathbb{Q} \Vdash x \in A_k$$

Then we define force with  $\text{Code}(x, m, k, 0, \eta)$ , provided  $F(\beta)_1 = \eta$  and  $\eta$  has not been used for coding yet. Otherwise we use  $\text{Code}(x, m, k, 0, \zeta)$  for  $\zeta$  the least ordinal which has not been used for coding yet.

If both (a)(i) and (a)(ii) are true for the same  $\zeta$ , then we give case (a)(i) preference, and suppress case (a)(ii).

(b) Else  $F$  guesses whether we code  $(x, m, k, 0)$  or  $(x, m, k, 1)$ , i.e. we code  $(x, m, k, F(\beta)_1)$  provided  $F(\beta)_1 \in 2$  (otherwise we decide to code  $(x, m, k, 0)$  into  $\vec{S}$  per default).

(c) If  $F(\beta)_0$  is of the form associated to the cases (2), (3), (4) and (5) in the definition of 0-allowable forcing, we proceed as described there. Thus  $\alpha$ -allowable forcings only act on case (1) in the definition of 0-allowable forcings, and leaves the other cases untouched.

This ends the definition of  $\mathbb{P}$  being  $\alpha$ -allowable with respect to  $F$  at successor stages  $\beta + 1$ . To define the limit stages  $\beta$  of an  $\alpha$ -allowable forcing, we assume that we have defined already  $(\mathbb{P}_\gamma : \gamma < \beta)$  and let the limit  $\mathbb{P}_\beta$  be defined as the direct limit as we use finite support.

We finally have finished the definition of an  $\alpha$ -allowable forcing relative to a perviously fixed bookkeeping function  $F$ . In the following we often drop the reference to  $F$  and simply say that some forcing  $\mathbb{P}$  is  $\alpha$ -allowable, in which case we always mean that there is some  $F$  such that  $\mathbb{P}$  is  $\alpha$ -allowable relative to  $F$ .

We briefly describe a typical run through the cases in the definition of  $\alpha$ -allowable forcings. Given our bookkeeping  $F : \delta \rightarrow H(\omega_2)^2$ , the according allowable  $\mathbb{P} = (\mathbb{P}(\beta) : \beta < \delta)$  forcing is constructed such that at every stage

$\beta < \delta$  we ask whether there exists for  $\zeta = 0$  a  $\mathbb{Q}$  such that (a)(i) becomes true. If not then we ask the same question for (a)(ii). If both are false, we pass to  $\zeta = 1$ , and so on. If (a) (i) or (a) (ii) never applies for any  $\zeta < \alpha$ , we pass to (b). It is therefore intuitively clear, and will be proved in a moment, that the notion of  $\alpha$ -allowable has to satisfy more and more requirements as  $\alpha$  increases, hence the classes of  $\alpha$ -allowable forcings should become smaller and smaller. As a further consequence of this, case (a) in the definition becomes easier and easier to satisfy, which leads in turn to more restrictions of how an  $\alpha$ -allowable forcing can look like. Next we list the main properties of  $\alpha$ -allowable forcings, all proof can be found in [5] again.

**Lemma 5.2.** *Let  $\beta < \alpha$  be ordinals.*

- *The notion  $\alpha$ -allowable is definable over the universe  $W$ .*
- *If  $\mathbb{P}$  is  $\beta$ -allowable then  $\mathbb{P}$  is also  $\alpha$ -allowable. Thus the classes of  $\alpha$ -allowable forcings become smaller with respect to the subset relation, if  $\alpha$  increases.*
- *let  $F_1, F_2$  be two bookkeeping functions,  $F_1 : \delta_1 \rightarrow W^2, F_2 : \delta_2 \rightarrow W^2$ , and let  $\mathbb{P}^1 = (\mathbb{P}_\eta^1 : \eta < \delta_1)$  and  $\mathbb{P}^2 = (\mathbb{P}_\eta^2 : \eta < \delta_2)$  be the  $\alpha$ -allowable forcings one obtains when using  $F_1$  and  $F_2$  respectively. Assume further that the range of  $F_1(\eta)_1$  and the range of  $F_2(\eta)_1$  are disjoint, i.e.  $C^{\mathbb{P}^1} \cap C^{\mathbb{P}^2} = \emptyset$ . Then  $\mathbb{P} := \mathbb{P}^1 \times \mathbb{P}^2$  is  $\alpha$ -allowable over  $W$ , as witnessed by some  $F : (\delta_1 + \delta_2) \rightarrow W^2$ , which is definable from  $\{F_1, F_2\}$ .*
- *For any  $\alpha$ , the set of  $\alpha$ -allowable forcings is non-empty.*

As a direct consequence of the last two observations we obtain that there must be an ordinal  $\alpha$  such that for every  $\beta > \alpha$ , the set of  $\alpha$ -allowable forcings must equal the set of  $\beta$ -allowable forcings. Indeed every allowable forcing is an  $\aleph_1$ -sized partial order, thus there are only set-many of them, and the classes (which in fact are sets, if we allow ourselves to identify isomorphic forcings) of  $\alpha$ -allowable forcings must eventually stabilize at a set which also must be non-empty.

**Definition 5.3.** *Let  $\alpha$  be the least ordinal such that for every  $\beta > \alpha$ , the set of  $\alpha$ -allowable forcings is equal to the set of  $\beta$ -allowable forcings. We say that some forcing  $\mathbb{P}$  is  $\infty$ -allowable if and only if it is  $\alpha$ -allowable. Equivalently, a forcing is  $\infty$ -allowable if it is  $\beta$ -allowable for every ordinal  $\beta$ .*

## 6 Thinning out while leaving out coding forcings

One of the main idea to construct the desired universe is to use the thinning out process detailed above, yet adding further information of which coding

forcings must not be used in the thinning out process. This idea will give us a very tight control over a certain  $\Pi_3^1$ -formula, even in the context of forcing  $\Pi_3^1$ -reduction, which is a rather fragile one. We will define the eventual iteration in a very careful way such that the mentioned  $\Pi_3^1$ -formula can not be uniformized by a  $\Pi_3^1$ -function. This refined thinning out process will simply replace the set of 0-allowable forcings, which form the basis of the old process with 0-allowable forcings which avoid some fixed set of (names of) reals  $B$ .

**Definition 6.1.** *Let  $B$  be an arbitrary set of tuples of (forcing names of) reals denoted by  $\vec{x}$ . We say that an iteration  $\{(\mathbb{P}_\alpha, \dot{\mathbb{Q}}_\alpha) \mid \alpha < \delta\}$  is 0-allowable without using  $B$  (or avoiding  $B$ ) if it is allowable and for every  $\alpha < \delta$  and every  $\mathbb{P}_\alpha$ -generic filter  $G_\alpha$ , none of the factors  $\dot{\mathbb{Q}}_\alpha^{G_\alpha}$  are of the form  $\text{Code}(\vec{x}^{G_\alpha}, \eta)$ ,  $\eta < \omega_1$  and  $\vec{x} \in B$ , where we write  $\vec{x}^{G_\alpha}$  for the evaluation of the  $\mathbb{P}_\alpha$ -names which are elements of  $\vec{x}$  with the help of the generic  $G_\alpha$ .*

We list useful properties of this new notion. The proofs are almost exactly the same as for plain allowable forcings so we skip them.

**Lemma 6.2.** *Let  $B$  be a set of pairs of reals. Let  $\delta$  be a countable ordinal and let  $F : \delta \rightarrow H(\omega_2)$  be a bookkeeping function. Finally let  $\mathbb{P} = ((\mathbb{P}_\alpha, \dot{\mathbb{Q}}_\alpha) \mid \alpha < \delta)$  be an allowable forcing avoiding  $B$  relative to  $F$ . Then  $\mathbb{P}$  has the following properties:*

- $\mathbb{P}$  has the ccc.
- $\mathbb{P}$  preserves CH.
- If  $\mathbb{Q}$  is a second allowable forcing avoiding  $B$  then  $\mathbb{P} \times \mathbb{Q}$  can be densely embedded into an allowable forcing avoiding  $B$ , provided  $C^\mathbb{P} \cap C^\mathbb{Q} = \emptyset$ .

It is straightforward to see that one can repeat the thinning out process detailed above yielding  $\infty$ -allowable forcings from the base set of 0-allowable forcings in exactly the same way if we start instead with 0-allowable forcings avoiding  $B$  as the base set of our forcings. To be more precise we can form the following. Inductively we assume that for an ordinal  $\alpha$  and an arbitrary bookkeeping function  $F \in W$  mapping to  $H(\omega_2)^2$ , we have already defined the notion of  $\delta$ -allowable avoiding  $B$  with respect to  $F$  for every  $\delta < \alpha$ , and the definition works uniformly for every model  $W[G]$ , where  $G$  is a generic filter for an allowable forcing. Note that these inductive requirements are met for 0-allowable forcings avoiding  $B$ . Now we aim to define the derivation of the set of  $\delta$ -allowable forcings avoiding  $B$  for  $\delta < \alpha$ . This will yield a smaller set of forcings which we call  $\alpha$ -allowable forcing avoiding  $B$ . In our iteration, the set  $B$  we want to avoid will never contain (names of) reals of the form  $(\dot{m}, \dot{k}, \dot{x})$  for  $\dot{m}, \dot{k}$  names for Gödel numbers of two  $\Pi_3^1$ -sets  $A_m$  and  $A_k$  and

$\dot{x}$  the name of a real. This will serve as a default assumption from now on which helps us ruling out some degenerate cases. <sup>4</sup>

**Definition 6.3.** *Let  $\zeta < \omega_1$  then a  $\zeta$ -length iteration  $\mathbb{P}$  is called  $\alpha$ -allowable avoiding  $B$  if it is recursively constructed using a bookkeeping function  $F : \delta \rightarrow H(\omega_2)^2$ , such that for every  $\beta < \zeta$ ,  $F(\beta)$  is a pair  $(F(\beta)_0, F(\beta)_1)$ , and two rules at every stage  $\beta < \zeta$  of the iteration. We assume inductively that we already created the forcing iteration up to  $\beta$ ,  $\mathbb{P}_\beta$  and we let  $G_\beta$  denote a hypothetical  $\mathbb{P}_\beta$ -generic filter over  $W$ . We shall now define the next forcing of our iteration  $\mathbb{P}(\beta) = \dot{\mathbb{Q}}_\beta^{G_\beta}$ . Using the bookkeeping  $F$  we split into two cases.*

- (a) *We assume first that the first coordinate of  $F(\beta)$ ,  $(F(\beta))_0 = (m, k, \dot{x})$ , where  $\dot{x}$  is the  $\mathbb{P}_\beta$ -name of a real and  $m < k$  are natural numbers. Further we assume that  $\dot{x}^{G_\beta} = x$ , and  $W[G_\beta] \models x \in A_m \cup A_k$ . We assume that in  $W[G_\beta]$ , the following is true:*

*There is an ordinal  $\zeta$ , which is chosen to be minimal for which*

- (i) *for every  $\zeta$ -allowable forcing avoiding  $B$ , dubbed  $\mathbb{Q} \in W[G_\beta]$  we have that, over  $W[G_\beta]$ :*

$$\mathbb{Q} \Vdash x \in A_m$$

*In this situation we force with  $\text{Code}(m, k, 1, x, \eta)$  for an ordinal  $\eta$  which did not appear yet as a coding area.*

- (ii) *If (i) for  $\zeta$  is false but the dual situation is true, i.e. for every  $\zeta$ -allowable forcing avoiding  $B$ , called  $\mathbb{Q} \in W[G_\beta]$ , we have that  $W[G_\beta]$  thinks that*

$$\mathbb{Q} \Vdash x \in A_k$$

*Then we define force with  $\text{Code}(m, k, 0, x, \eta)$ , for a free  $\eta$ .*

*If both (a)(i) and (a)(ii) are true for the same  $\zeta$ , then we give case (a)(i) preference, and suppress case (a)(ii).*

- (b) *If (a) (i) and (a) (ii) are both false, then  $F$  guesses where we code  $(x, m, k)$ , i.e. we code  $(x, m, k)$  into  $\bar{S}^{F(\beta)_1}$ , provided  $F(\beta)_1 \in 2$  (otherwise we decide to code  $(x, m, k)$  into  $\bar{S}^1$  per default).*

---

<sup>4</sup>Otherwise the situation could arise that the rules of  $\alpha$ -allowable want us to force with coding a tuple  $\vec{x}$  at some  $\eta$ , which can cause trouble if a name of that tuple happens to belong to  $B$ . Our assumption rules this possibility out from the very beginning in demanding that no element of  $B$  is of a form which would enable such an unwanted situation.

This ends the definition of  $\mathbb{P}$  being  $\alpha$ -allowable with respect to  $F$  avoiding  $B$  at successor stages  $\beta + 1$ . To define the limit stages  $\beta$  of an  $\infty + \alpha$ -allowable forcing avoiding  $B$ , we assume that we have defined already  $(\mathbb{P}_\gamma : \gamma < \beta)$  and let the limit  $\mathbb{P}_\beta$  be defined as the direct limit as we use finite support.

The properties of  $\alpha$ -allowable forcings carry over to  $\alpha$ -allowable forcings avoiding  $B$ . The proofs are almost the same, all we need to do is to replace every instance of “ $\alpha$ -allowable” with “ $\alpha$ -allowable avoiding  $B$ ”.

**Lemma 6.4.** *Let  $\beta < \alpha$  be ordinals.*

- *The notion  $\alpha$ -allowable avoiding  $B$  is definable over the universe  $W$ .*
- *If  $\mathbb{P}$  is  $\beta$ -allowable avoiding  $B$  then  $\mathbb{P}$  is also  $\alpha$ -allowable avoiding  $B$ . Thus the classes of  $\alpha$ -allowable forcings avoiding  $B$  become smaller with respect to the subset relation, if  $\alpha$  increases.*
- *let  $F_1, F_2$  be two bookkeeping functions,  $F_1 : \delta_1 \rightarrow W^2, F_2 : \delta_2 \rightarrow W^2$ , and let  $\mathbb{P}^1 = (\mathbb{P}_\eta^1 : \eta < \delta_1)$  and  $\mathbb{P}^2 = (\mathbb{P}_\eta^2 : \eta < \delta_2)$  be the  $\alpha$ -allowable forcings avoiding  $B$  one obtains when using  $F_1$  and  $F_2$  respectively. Assume further that the range of  $F_1(\eta)_1$  and the range of  $F_2(\eta)_1$  are disjoint, in other words that  $C^{\mathbb{P}^1} \cap C^{\mathbb{P}^2} = \emptyset$ . Then  $\mathbb{P} := \mathbb{P}^1 \times \mathbb{P}^2$  is  $\alpha$ -allowable avoiding  $B$  over  $W$ , as witnessed by some  $F : (\delta_1 + \delta_2) \rightarrow W^2$ , which is definable from  $\{F_1, F_2\}$ .*
- *For any  $\alpha$ , the set of  $\alpha$ -allowable forcings avoiding  $B$  is non-empty.*

The set of  $\alpha$ -allowable forcings avoiding  $B$  will stabilize at a non-empty set of forcings, just as before.

**Definition 6.5.** *Let  $B$  be a set of pairs of reals. The set of  $\infty$ -allowable forcings avoiding  $B$  is the non-empty result of repeating the derivation detailed above until we reach a fixed point, i.e. until we reach an ordinal  $\alpha$  such that the notion of  $\alpha$ -allowable avoiding  $B$  coincides with the notion of  $\alpha + 1$ -allowable avoiding  $B$ . We note that, as there are only set-many partial orders of size  $\aleph_1$ , modulo isomorphism, such an ordinal  $\alpha$  must exist.*

Having obtained the fixed point for the thinning out process avoiding  $B$ , one can argue for forcing  $\Pi_3^1$ -reduction as follows. We will define an  $\omega_1$ -length iteration such that every initial segment of the iteration is  $\infty$ -allowable forcings avoiding  $B$ . Suppose we are at stage  $\beta < \omega_1$  of the iteration and the bookkeeping  $F$  is considering a real  $x$  and two  $\Pi_3^1$ -set  $\varphi_m$  and  $\varphi_k$ . In order to avoid trivialities we assume that  $x$  is an element of  $A_m$  and  $A_k$  at our current stage of the iteration. There are three cases:

- Our real  $x$  can not be forced out of  $A_m$  with an  $\infty$ -allowable forcing avoiding  $B$ . In this situation we force to put  $x$  into  $D_{m,k}^0$ , the set which should eventually become a subset of  $A_m$ .

- If not, then we assume that our real  $x$  can not be forced out of  $A_k$  with an  $\infty$ -allowable forcing avoiding  $B$ . Then we force to put  $x$  into  $D_{m,k}^1$  the set which will become a subset of  $A_k$ .
- Finally  $x$  can be forced out of  $A_m$  with a forcing  $\mathbb{P}_0$  which is  $\infty$ -allowable avoiding  $B$ ; and  $x$  can also be forced out of  $A_k$  with  $\mathbb{P}_1$  which is  $\infty$ -allowable avoiding  $B$ . In this situation we use the product  $\mathbb{P}_0 \times \mathbb{P}_1$  which is  $\infty$ -allowable avoiding  $B$  and which, by upwards absoluteness of  $\Sigma_3^1$ -formuals, forces  $x$  out of  $A_m \cup A_k$ .

The so defined iteration results in an  $\infty$ -allowable forcing avoiding  $B$  again, hence all the placements of reals are valid and the  $\Pi_3^1$ -reduction property holds as soon as we took care of all the reals in our universe, which is the case if we iterate of length  $\omega_1$  using a suitable bookkeeping.

We end this section with a brief outlook of how our iteration will look like which will force the main theorem. Having defined  $\infty$ -allowable forcings avoiding  $B$ , we will continue in the following fashion: First we force with a countable length iteration of  $\infty$ -allowable forcings avoiding  $B_0$  arriving at some  $W[G_0]$ . In  $W[G_0]$  we will define a new set of tuples  $B_1$  we want to avoid. Then, working over  $W[G_0]$ , we start a new thinning out process with the set of  $\infty$ -allowable forcings avoiding  $B_0$  and  $B_1$  as the base set of our thinning out process. This process will stabilize after infinitely many thinning out stages and yield a class we call for obvious reasons the set of  $\infty + \infty = \infty \cdot 2$ -allowable forcings avoiding  $(B_0, B_1)$ . Then we force over  $W[G_0]$  with such a  $\infty \cdot 2$ -allowable forcing which avoids  $(B_0, B_1)$  and arrive at a universe  $W[G_0][G_1]$ . Over this model we single out a set  $B_2$  of reals we want to avoid, start a new thinning out process and arrive at the notion of  $\infty \cdot 3$ -allowable forcings which avoid  $(B_0, B_1, B_2)$  and force with such a forcing. This process can be iterated again transfinitely often yielding  $\infty \cdot \alpha$  allowable forcings avoiding  $(B_\eta \mid \eta < \alpha)$  which is what we will do in the proof of the main theorem.

## 7 Proof of the main theorem

We will start to define the iteration that will prove the main theorem now. There are four different tasks we have to take care of:

1. Forcing a failure of  $\Pi_3^1$ -uniformization,
2. forcing the  $\Pi_3^1$ -reduction,
3. forcing the  $\Sigma_4^1$ -uniformization property and
4. forcing a good  $\Sigma_5^1$ -wellorder of the reals.

The organization of the proof will prioritize establishing the failure of  $\Pi_3^1$ -uniformization before addressing the remaining three properties. The initial iteration is defined over the ground model  $W$ ; however, we will promptly transition to an intermediate model,  $W[G_\omega]$ . This intermediate model possesses the critical feature that  $\Pi_3^1$ -uniformization fails within it, and this failure is maintained in certain carefully specified outer models of  $W[G_\omega]$ . Working over the intermediate model  $W[G_\omega]$ , we proceed to construct an  $\omega_1$ -length iteration, employing the “stop and go” method detailed at the conclusion of the preceding section. This method involves an alternating sequence of steps: first, a new set of names for reals that must be avoided is defined; second, a thinning-out process is initiated with allowable forcings avoiding additionally this new set of names of reals playing the role of our base set, until a fixed point is attained. Upon reaching this fixed point, a forcing is applied from this fixed point to extend our iteration. Then a new set of names for reals to be additionally avoided is defined, subsequently a new thinning-out process is begun until the next fixed point is reached, again we will use a forcing from this new fixed point, then halt again, and this cycle continues.

## 7.1 Forcing the failure of $\Pi_3^1$ -uniformization

As mentioned already the goal is to create first a universe where the  $\Pi_3^1$  uniformization property fails and more importantly continues to fail in all outer models which are obtained using a certain, carefully defined set of forcings. The plan is then to work towards  $\Pi_3^1$ -reduction,  $\Sigma_4^1$ -uniformization and a good  $\Sigma_5^1$  well-order of the reals with forcings which belong to this carefully defined set of forcings. Hence  $\Pi_3^1$ -uniformization continues to fail in these outer models.

The basic first idea is to consider the set

$$A := \{(x, y) \in (\omega^\omega)^2 \mid (0, 0, x, y) \text{ is not coded into } \vec{S}\}$$

which is  $\Pi_3^1$ .

We will employ our coding forcings to construct a universe where the set  $A$  cannot be uniformized by any  $\Pi_3^1$ -function, thus demonstrating the failure of  $\Pi_3^1$ -uniformization. This objective is in considerable tension with the simultaneous goal of achieving  $\Pi_3^1$ -reduction. This tension, however, can be mitigated if we use the coding forcings of the form  $\text{Code}(n, x, \eta)$  for  $n \in \omega$  from case (5) in the definition 4.1 of 0-allowable forcings. These forcings will be used as some sort of yardstick which will mark potentially dangerous places for codes. The idea is inspired by a similar construction in [4].

### 7.1.1 Definition of the first iteration in general

We shall define a general version of the construction now and soon apply it under more specific circumstances. We let  $F : \omega_1 \rightarrow H(\omega_1)$  be some

bookkeeping function that should have the property that every element of  $H(\omega_1)$  has uncountable preimage. Suppose that we are at stage  $\beta < \omega_1$  of our to-be-defined iteration. We further assume that our iteration up to  $\beta$ ,  $\mathbb{P}_\beta$  and a  $\mathbb{P}_\beta$ -generic filter  $G_\beta$  are already defined. We also assume that we have already defined a set  $\{B_m \mid m \leq n\}$  where each  $B_m$  is a set of names of reals that we want to avoid with our 0-allowable forcing. We look at the  $F$ 's value at stage  $\beta$

$$F(\beta) = (m, x), \text{ where } x \in \omega^\omega \text{ and } m \in \omega.$$

We shall split into two cases.

1. We assume first that  $x$  has not been considered before by the bookkeeping function. We also assume that

$$\begin{aligned} W[G_\beta] \models \exists \mathbb{P} \exists y \in \omega^\omega (\mathbb{P} \text{ is 0-allowable avoiding } \bigcup_{m \leq n} B_m \text{ and} \\ \forall \eta \forall \dot{R} (\text{Code}(n, \dot{R}, \eta) \text{ is not a factor of } \mathbb{P}) \text{ and} \\ \mathbb{P} \Vdash (x, y) \notin A_m) \end{aligned}$$

In this situation, we first fix the  $<$ -least such allowable  $\mathbb{P}$  and use it at stage  $\beta$  of our iteration, that is, we let

$$(\dot{Q}_\beta^0)^{G_\beta} = \mathbb{P}.$$

In a second step we let  $R_n$  be a real which codes all the generic reals we have created so far.<sup>5</sup> Then, if  $H^0$  denotes a  $\mathbb{P}$ -generic filter over  $W[G_\beta]$  and working over  $W[G_\beta][H^0]$  we force with

$$(\dot{Q}_\beta^1)^{G_\beta} := \text{Code}(n, R_n, \eta)$$

for an  $\eta$  which is free for coding. We let

$$B_{n+1} := \{\dot{z}, n \mid \dot{z} \text{ a name of a real}\} \cup \{(x, y)\}$$

We settle to only use 0-allowable forcings which avoid  $\bigcup\{B_m \mid m \leq n + 1\}$  from now on. in other words we decide to not use a coding forcing of the form  $\text{Code}(n, z, \eta)$  ever again and also avoid a coding forcing of the form  $\text{Code}(0, 0, x, y, \eta)$ .

---

<sup>5</sup>Note that allowable forcings are just a countable length iteration of almost disjoint coding forcings. Hence each allowable extension of  $W$  can be written as  $W[R]$  for one real  $R$  which codes all the a.d. reals added so far.

2. We assume that case 1 does not apply. In particular there is no allowable forcing  $\mathbb{P}$  avoiding  $\bigcup\{B_m \mid m \leq n\}$  which forces a pair  $(x, y)$  out of  $A_m$  without having a coding forcing  $\text{Code}(n, \dot{R}, \eta)$  and without having  $\text{Code}(x, y, \eta)$  as a factor. In this situation we opt to never use a coding forcing of the form  $\text{Code}(n, \dot{R}, \eta)$  and  $\text{Code}(x, y, \eta)$  ever again as a factor. That is we form

$$B_{n+1} = \{(n, \dot{R}) \mid \dot{R} \text{ a name for a real}\} \cup \{(x, \dot{y}) \mid \dot{y} \text{ a name of a real}\}$$

and use allowable forcings avoiding  $\bigcup\{B_m \mid m \leq n + 1\}$  from now on. We do not force at this stage.

3. If the real  $x$  has already been considered by the bookkeeping function at an earlier stage  $\gamma < \beta$ , and if at stage  $\gamma$  case 1 applied and a real  $y$  has been singled out such that  $\{(x, y)\}$  is an element of  $\bigcup_{m \leq n} B_m$ , then we pick a real  $y' \neq y$ , which has not been considered yet and use

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(x, y', \eta)$$

for some  $\eta < \omega_1$  which has not been used for coding yet.

4. If the real  $x$  has been considered earlier in our iteration and case 2 applied there, then we do not force.

If we iterate of length  $\omega_1$  following the four rules in a way such that each pair of reals is considered cofinally often by our bookkeeping and let  $W[G_{\omega_1}]$  denote the resulting universe and let  $B_\omega := \bigcup_{m < \omega} B_m$ , then the universe will satisfy the following.

**Lemma 7.1.** *Let  $W[G_{\omega_1}]$ ,  $B_\omega$  be as just specified.*

1. *Then*

$$A := \{(x, y) \in (\omega^\omega)^2 \mid (0, 0, x, y) \text{ is not coded into } \vec{S}\}$$

*is a  $\Pi_3^1$ -set which can not be uniformized by any  $\Pi_3^1$ -graph.*

2. *For any further outer universe  $\tilde{W}$  of  $W[G_{\omega_1}]$  which is obtained via an allowable forcing avoiding  $B_\omega$ , there is a further allowable forcing avoiding  $B_\omega$   $\mathbb{Q}$  over  $\tilde{W}$  such that in  $\tilde{W}^\mathbb{Q}$  the  $\Pi_3^1$ -uniformization property fails.*

*Proof.* To prove the first part we let  $m$  be a natural number which is the Gödel number of a  $\Pi_3^1$ -formula  $\varphi_m$  in two free variables. Let  $A_m$  denote the  $\Pi_3^1$  set of reals associated with  $\varphi_m$ . Suppose that  $\beta$  is the first stage such that there is a real  $x \in W[G_{\omega_1}]$  and  $F(\beta) = (m, \dot{x})$ .

If case 1 in the definition of our iteration applied at stage  $\beta$  and  $y$  is the real witnessing this, then  $A_m$  will not contain  $(x, y)$  by the definition of case 1, yet  $(x, y)$  is the unique element of  $A$  at its  $x$ -section as computed in  $W[G_{\omega_1}]$  as we assumed that every  $(x, y'), y' \neq y$  is coded. So  $A_m$  can not uniformize  $A$  in  $W[G_{\omega_1}]$ .

If case 2 applied at stage  $\beta$  then for any real  $y \in W[G_\beta]$ ,  $(x, y) \in A_m$  as computed in  $W[G_{\omega_1}]$ , so  $A_m$  is not even the graph of a function.

So  $\Pi_3^1$ -uniformization fails in  $W[G_{\omega_1}]$  as claimed.

To prove the second part, starting with an arbitrary  $\tilde{W}$ , we just need to ensure with an allowable forcing avoiding  $B_\omega \mathbb{Q}$  that eventually every real gets considered by the bookkeeping. This is straightforward to do.  $\square$

### 7.1.2 A first $\omega$ -length iteration

We next will define an  $\omega$ -length iteration of allowable forcings avoiding more and more sets as we advance in the iteration. The final model will have the property that  $\Pi_3^1$ -uniformization will fail there. Additionally it will be possible to define an iteration on top of this model which will secure  $\Pi_3^1$ -reduction while  $\Pi_3^1$ -uniformization continues to fail. We start with  $W$  as the ground model and start our consideration with the set of allowable forcings. We let  $B_0$  be the empty set and assume that we list all  $\Pi_3^1$ -formulas with two free variables  $(\varphi_m \mid m \in \omega)$  in such a way that case 1 and case 2 are alternating, starting with case 1 at the very first stage of our iteration. Note that such an assumption is harmless, as there are infinitely many formulas where case 1 must always apply and likewise there are infinitely many formulas where case 2 must apply as well, thus we can always re-organize any enumeration  $(\varphi_m \mid m \in \omega)$  in a way which meets the requirement. We proceed in the iteration for  $\omega$ -many steps as detailed in the definition of the first two cases above and let  $W[G_\omega]$  be the resulting model. Note that in  $W[G_\omega]$ , all  $\Pi_3^1$ -formulas with two free variables have been considered at some point by the bookkeeping.

**Lemma 7.2.** *For a real  $R$ , let  $(R)_n$  denote the  $n$ -th part of a recursively definable partition of  $R$  into  $\omega$ -many parts. In  $W[G_\omega]$  the following  $\Pi_2^1$ -formula holds true for exactly one real  $R$*

$$\Psi(R) \equiv (0, (R)_0) \text{ is coded} \wedge \forall n \in \omega (2n, (R)_{2n}) \text{ is coded} \wedge (R)_{2n} \notin L[(R)_{2n-2}]$$

*Proof.* We work in  $W[G_\omega]$ . First we note that there is exactly one real  $R_0$  such that  $(0, R_0)$  is coded, by the first case of the definition of the iteration. Now whenever we were at an odd stage of our iteration, we were in case 2 of the definition, thus we did not force at all and just defined a new set  $B_{n+1}$  and a new notion of  $\infty$ -allowable avoiding  $(B_m \mid m \leq n + 1)$ . Whenever we

hit an even stage  $2n$  of the iteration, we first kick a pair  $(x, y)$  out of the  $\Pi_3^1$ -set  $A_m$  handed to us by the bookkeeping and let  $R_{2n}$  be a real which codes the countably many reals which fully determine the iteration so far. Additionally we use  $\text{Code}(2n, R_{2n}, \eta)$  and by definition of the iteration  $R_{2n}$  it is the unique real which is not an element of  $L[R_{2n-2}]$  and for which  $(2n, R_{2n})$  is coded. To summarize, the real  $R$  which codes the sequence  $(R_{2n} \mid n \in \omega)$  is the unique real satisfying the  $\Pi_2^1$ -formula as desired.  $\square$

## 7.2 Preliminary remarks for the second iteration

We shall work now towards the three other properties our universe eventually should satisfy. This will be achieved using an  $\omega_1$ -length iteration over the ground model  $W[G_\omega]$ . We note that  $W[G_\omega]$  has a set of reals  $B_\omega$  associated which we must not use. As  $B_\omega$  will be the first in an infinite sequence of sets of (names of reals) we re-index and let

$$B_0 := B_\omega$$

from now on and hope that it will not confuse the reader. At each stage of the iteration we refine additionally the notion of allowable forcings we currently have. The iteration is defined by induction. It will have the crucial property that no matter what generic extension it will eventually produce, the  $\Pi_3^1$ -uniformization property will continue to fail there using the second item of lemma 7.1.1. We let  $F : \omega_1 \rightarrow H(\omega_1)$  be the bookkeeping function which organizes the iteration. The choice of  $F$  does not matter as long as every element of  $H(\omega_1)$  has an uncountable pre-image under  $F$  (this assumption is more than enough for our needs). We detail our assumptions for the inductive definition of the iteration. Assume we are at stage  $\beta$  of our iteration, let  $G_\beta$  be the  $\mathbb{P}_\beta$ -generic filter and we work over the universe  $W[G_\omega][G_\beta]$ . We also assume that for every  $\delta \leq \beta$  we have defined a notion of  $\infty \cdot \delta$ -allowable forcings avoiding  $(B_\eta \mid \eta \leq \delta)$ . The base case of the induction is the notion of  $(0)$ -allowable avoiding  $B_0$ , a notion we have defined over  $W[G_\omega]$ . We also set  $B_1 := B_0$  and define the notion of  $\infty \cdot 1$ -allowable avoiding  $(B_0)$  as detailed in Definition 6.5. We distinguish three cases and will work through them in the next three subsections.

## 7.3 Forcing $\Pi_3^1$ -reduction

We first assume that  $F(\beta) = (\beta_1, \beta_2)$  and that the  $\beta_1$ -th (in some previously fixed well-order  $<$  of  $H(\omega_2)$ )  $\mathbb{P}_{\beta_0}$ -name of a triple of the form  $(\dot{n}, \dot{l}, \dot{a})$ , where  $\dot{a}$  is a nice  $\mathbb{P}_{\beta_0}$ -name of a real, and  $\dot{n}, \dot{l}$  are nice  $\mathbb{P}_{\beta_0}$ -names of natural numbers, is the triple  $(\dot{m}, \dot{k}, \dot{x})$ . We let  $m = \dot{m}^{G_\beta}, k = \dot{k}^{G_\beta}$  and assume that  $m, k$  are both Gödel numbers of  $\Pi_3^1$ -formulas and  $x = \dot{x}^{G_\beta}$  is a real. In this situation we work towards  $\Pi_3^1$ -reduction. We work over  $W[G_\omega][G_\beta]$  as the ground

model. This model has associated a sequence of sets of tuples of (names of) reals

$$B := (B_\eta \mid \eta < \beta)$$

and, for every  $\eta \leq \beta$ , the notion of  $\infty \cdot \eta$ -allowable avoiding  $(B_i \mid i < \eta)$  as detailed earlier.

We distinguish several cases.

### 7.3.1 Forcing $\Pi_3^1$ -reduction, case 1

We assume that in the universe  $W[G_\omega][G_\beta]$  it is true that

$$W[G_\omega][G_\beta] \models \forall \mathbb{Q} (\mathbb{Q} \text{ is } \infty \cdot \beta\text{-allowable avoiding} \\ (B_i \mid i < \beta) \rightarrow \mathbb{Q} \Vdash x \in A_m),$$

then force with

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, k, 1, x, \eta).$$

Note that this has the direct consequence that if we restrict ourselves from now on to forcings  $\mathbb{Q} \in W[G_\omega][G_{\beta+1}]$  such that  $\mathbb{Q}$  is  $\infty \cdot \zeta_0$ -allowable avoiding  $(B_i \mid i < \zeta_0)$ , for  $\zeta_0 > \beta$ , then  $x$  will remain an element of  $A_m$ . In particular, the pathological situation that  $x \notin A_m$ ,  $x \in A_k$  while  $x$  is coded into  $\vec{S}^1$  is ruled out for  $(m, k, x)$ .

### 7.3.2 Forcing $\Pi_3^1$ -reduction, case 2

We assume that case 1 does not hold however we assume that in the universe  $W[G_\omega][G_\beta]$ ,

$$W[G_\omega][G_\beta] \models \forall \mathbb{Q} (\mathbb{Q} \text{ is } \infty \cdot \beta\text{-allowable avoiding} \\ (B_i \mid i < \beta) \rightarrow \mathbb{Q} \Vdash x \in A_k),$$

then force with

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, k, 0, x, \eta).$$

### 7.3.3 Forcing $\Pi_3^1$ -reduction, case 3

In the final case we assume that neither case 1 nor case 2 applies. We obtain that there is a forcing  $\mathbb{Q}$  with

$$W[G_\omega][G_\beta] \models \mathbb{Q} \text{ is } \infty \cdot \beta\text{-allowable avoiding } (B_i \mid i < \beta) \text{ and} \\ \mathbb{Q} \Vdash x \notin A_m.$$

Likewise we also obtain that there is a  $\mathbb{R}$

$$W[G_\omega][G_\beta] \models \mathbb{R} \text{ is } \infty \cdot \beta\text{-allowable avoiding } (B_i \mid i < \beta) \text{ and} \\ \mathbb{R} \Vdash x \notin A_k.$$

In particular there are  $\infty \cdot \beta$ -allowable forcings avoiding  $(B_i \mid i < \beta)$  which kick  $x$  out of  $A_m$  and  $A_k$  respectively. We do want to force  $x$  out of  $A_m$  and  $A_k$  but have to be a bit careful.

Given an  $\infty \cdot \beta$ -allowable forcing avoiding  $(B_\eta \mid \eta < \beta)$   $\mathbb{R} = (\mathbb{R}_\eta, \dot{\mathbb{Q}}_\eta)$  We define an  $\infty \cdot \beta$ -allowable forcing avoiding  $(B_\eta \mid \eta < \beta)$   $C^\omega(\mathbb{R})$ , which we call the closure of  $\mathbb{R}$  in the following way. Whenever  $\beta$  is a stage of  $\mathbb{R} = (\mathbb{R}_\eta, \dot{\mathbb{Q}}_\eta)$  where a real  $a$  and  $A_m, A_k$  are considered by the bookkeeping associated with  $\mathbb{R}$ , and  $a$  can always be kicked out of  $A_m$  and  $A_k$  with an  $\infty \cdot \beta$ -allowable forcing  $\mathbb{R}'$  avoiding  $(B_\eta \mid \eta < \beta)$  we use

$$C^1(\dot{\mathbb{Q}}_\beta) := \dot{\mathbb{Q}}_\beta \times \mathbb{R}'$$

instead of just  $\dot{\mathbb{Q}}_\beta$  at this stage of the iteration. Iterating the  $C^1(\dot{\mathbb{Q}}_\beta)$ 's with finite support results in the first step of the closure procedure we call  $C^1(\mathbb{R}) := \ast C^1(\dot{\mathbb{Q}}_\beta)$ . We now iterate this  $\omega$ -often, that is we form  $C^{n+1}(\mathbb{R}) := C(C^n(\mathbb{R}))$  and let  $C^\omega(\mathbb{R})$  be the direct limit of these forcings. The result  $C^\omega(\mathbb{R})$  is the closure of  $\mathbb{R}$ . We state some properties which follow immediately from its definition.

**Lemma 7.3.** *Let  $\mathbb{R}$  be an  $\infty \cdot \beta$ -allowable forcing avoiding  $(B_\eta \mid \eta < \beta)$  relative to the bookkeeping  $F'$  and let  $C^\omega(\mathbb{R})$  be its closure. Then  $C^\omega(\mathbb{R})$  is itself  $\infty \cdot \beta$ -allowable avoiding  $(B_\eta \mid \eta < \beta)$  relative to some  $F''$  and will have the property that whenever there is an  $\eta$  such that  $F''(\eta) = (m, k, a)$  and  $a$  can both be forced out of  $A_m$  and  $A_k$  with an  $\infty \cdot \beta$ -allowable forcing, then  $C^\omega(\mathbb{R}) \Vdash x \notin A_m \cup A_k$ .*

Returning to our iteration at stage  $\beta$ , we now let  $C^\omega(\mathbb{Q})$  and  $C^\omega(\mathbb{R})$  be the closure of the  $<$ -least  $\infty \cdot \beta$ -allowable forcings avoiding  $(B_i \mid i < \beta)$  as above and use

$$\dot{\mathbb{Q}}^{G_\beta} := C^\omega(\mathbb{Q}) \times C^\omega(\mathbb{R})$$

which is an  $\infty \cdot \beta$ -allowable forcing avoiding  $(B_i \mid i < \beta)$  over  $W[G_\omega][G_\beta]$  and which forces that  $x \notin A_m \cup A_k$ .

For all three cases above we finally let  $B_\beta := \emptyset$  and define  $\infty \cdot (\beta + 1)$ -allowable avoiding  $(B_\eta \mid \eta < \beta + 1)$  as being just the same notion as  $\infty \cdot \beta$  avoiding  $(B_\eta \mid \eta < \beta)$ . This ends the definition of the iteration in this case and we shall show that, if  $G_{\omega_1}$  denotes a generic filter for the forcing  $\mathbb{P}_{\omega_1}$ , which is defined as the direct limit of the forcings  $\mathbb{P}_\beta$ , then the resulting universe  $W'[G_{\omega_1}]$  satisfies the  $\Pi_3^1$ -reduction property.

## 7.4 Forcing $\Sigma_4^1$ -uniformization

This section deals with forcing the  $\Sigma_4^1$ -uniformization property. First we note that we need to define how to force  $\Sigma_4^1$ -uniformization in a different way to [7]. This is necessary, as the latter machinery can not be combined

with forcing  $\Pi_3^1$ -reduction. Indeed both methods exclude themselves and so something new is in demand. For every integer  $n \in \omega$  we define the set

$$f_n := \{(x, y) \mid \exists a_0 \forall a_1 ((n, x, y, a_0, a_1) \text{ is not coded into } \vec{S})\}$$

and will work towards a universe where for every  $n \in \omega$ , if  $n$  is the Gödel number of a  $\Sigma_4^1$ -set  $A_n$  in the plane then  $f_n$  uniformizes  $A_n$ . Note that  $f_n$  is a  $\Sigma_4^1$ -formula, hence  $\Sigma_4^1$ -uniformization would hold.

Assume that  $F(\beta)$  hands us a Gödel number  $m$  of a  $\Sigma_4^1$ -formula

$$\varphi_m = \exists a_0 \forall a_1 \psi_m(x, y, a_0, a_1)$$

where  $\psi_m$  is a  $\Sigma_2^1$ -formula, hence absolute by Shoenfield's theorem. Also assume that  $F(\beta)$  determines reals  $x, y', a'_0$ . We assume that our uniformizing function  $f_m$  for  $\varphi_m$  is not yet defined at  $x$ . Additionally we assume that inductively we have defined for every  $\eta \leq \beta$  a notion of  $\infty \cdot \eta$ -allowable avoiding  $(B_\gamma \mid \gamma < \eta)$ . Let  $G_\beta$  be a  $\mathbb{P}_\beta$ -generic filter over  $W[G_\omega]$ . We work in  $W[G_\omega][G_\beta]$ .

#### 7.4.1 Case 1

We ask at this stage whether there is a real  $y$  and a  $\infty \cdot \beta$ -allowable forcing  $\mathbb{R} = ((\mathbb{R}_\eta, \dot{Q}_\eta) \mid \eta < \beta)$  avoiding  $(B_\eta \mid \eta < \beta)$  which adds a real  $a_0$  such that for all further  $\infty \cdot \beta$ -allowable forcings  $\mathbb{R}'$  avoiding  $(B_\eta \mid \eta < \beta)$  it is true that

$$\mathbb{R}' \Vdash \forall a_1 \psi_m(x, y, a_0, a_1).$$

In this situation we fix the  $<$ -least such real  $y = f_m(x)$  as the value of our eventual uniformizing function at  $x$ . We would want to add such a real  $a_0$  with such a forcing  $\mathbb{R}$  but we need to be careful. We also have to ensure that our work towards  $\Pi_3^1$ -reduction will not cause problems.

In order to circumvent this difficulty we again “close the forcing  $\mathbb{R}$  off”. So instead of using a forcing  $\mathbb{R}$  which introduces the real  $a_0$  at stage  $\beta$  of our iteration, we force with its closure

$$\dot{Q}_\beta^{G_\beta} := C^\omega(\mathbb{R}).$$

Note that  $C^\omega(\mathbb{R})$  also adds the real  $a_0$ , but, as already discussed, circumvents potential difficulties when dealing with  $\Pi_3^1$ -reduction.

Last we generically add a fresh real  $c_0(x, y, m)$  via using a coding forcing which codes some harmless information <sup>6</sup> and define

$$B_\beta := \{(m, x, y, c_0(x, y, m), \dot{z}) \mid \dot{z} \text{ some name of a real}\}$$

---

<sup>6</sup>For example, if  $l$  is the Gödel number of a  $\Sigma_4^1$ -set which is always empty then we do not need to uniformize  $A_l$ , yet we can use coding forcings of the form  $\text{Code}(l, a, b)$  for arbitrary reals  $a, b$  which would produce such a desired fresh real.

and let the notion of  $\infty \cdot (\beta + 1)$ -allowable avoiding  $(B_\eta \mid \eta < \beta + 1)$  be defined as starting the infinite thinning out process with the base set of  $\infty \cdot \beta$ -allowable forcings which avoid  $(B_\eta \mid \eta < \beta)$  which additionally avoid  $B_\beta$ . The resulting fixed point will be dubbed  $\infty \cdot (\beta + 1)$ -allowable forcings avoiding  $(B_\eta \mid \eta < \beta + 1)$ . This definition implies that we must not use a forcing of the form  $\text{Code}(m, x, y, c_0(x, y, m), z, \eta)$  for any  $\eta$  and any real  $z$  which will ensure that throughout our iteration

$$“(m, x, y, c_0(x, y, m), z) \text{ is not coded}”$$

will remain true. Note that this is a  $\Pi_3^1$ -property in the parameters  $(m, x, y, c_0(m, x, y), z)$ , hence asserting that there is a real  $c_0(m, x, y)$  which witnesses that “ $(m, x, y, c_0(x, y, m), z)$  is not coded” becomes a  $\Sigma_4^1$ -property in the parameters  $(x, y)$ , as we can define the integer  $m$ .

We finally demand that from now on, whenever we visit  $x$  and  $\varphi_m$  again in our iteration, we use a coding forcing  $\text{Code}(m, x, y', a_0, a_1)$ . This will ensure that eventually all other reals  $y' \neq y = f_m(x)$  will satisfy “ $\exists a_0 \forall a_1 ((m, x, y', a_0, a_1)$  is coded”. So  $y$  will become the unique real which satisfies the  $\Sigma_4^1$ -formula

$$\exists a_0 \forall a_1 ((x, y, a_0, a_1) \text{ is not coded})$$

hence we produce a uniformizing function at  $x$ .

#### 7.4.2 Case 2

If on the other hand, for any real  $y$  we add with an  $\infty \cdot \beta$ -allowable forcing avoiding  $(B_\eta \mid \eta < \beta)$  and any real  $a_0$  there is a further  $\infty \cdot \beta$ -allowable forcing avoiding  $(B_\eta \mid \eta < \beta)$ ,  $\mathbb{Q}'$  such that

$$\mathbb{Q}' \Vdash \exists a_1 \neg \psi_m(x, y', a'_0, a_1)$$

then we let  $F$  hand us the reals  $y'$  and  $a'_0$  and use the closure  $C^\omega(\mathbb{Q}')$  of such a suitable forcing  $\mathbb{Q}'$  at stage  $\beta$  of our iteration. To be more precise we let

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := C^\omega(\mathbb{Q}').$$

As a result  $\exists a_1 \neg \psi_m(x, y', a'_0, a_1)$  becomes true. Note that this is a  $\Sigma_3^1$ -formula, hence it remains true in all further generic extensions we will produce in our iteration.

We also keep our old notion of allowable, in other words we let  $B_\beta := \emptyset$  and let  $\infty \cdot \beta + 1$ -allowable avoiding  $(B_\eta \mid \eta < \beta + 1)$  be just our old  $\infty \cdot \beta$ -allowable avoiding  $(B_\eta \mid \eta < \beta)$ .

This ends the definition of our method of forcing the  $\Sigma_4^1$ -uniformization property.

## 7.5 Forcing a good $\Sigma_5^1$ -well-order

Our methods allow for an additional layer of complexity which we can use to force a good  $\Sigma_5^1$ -well-order of the reals. We single out coding forcings of the form

$$\text{Code}(1, 1, x, y, \eta)$$

to be our tool which eventually should yield the good  $\Sigma_5^1$ -well-order. The idea is to use the coding forcing  $\text{Code}(1, 1, x, z, \eta)$  in such a way that

$$(1, 1, x, z) \text{ is coded "cofinally often"} \Leftrightarrow \\ \text{"}z \text{ codes the initial segment of the well-order below } x\text{"}$$

Here being coded "cofinally often" is just an abbreviation for the formula " $\exists r_0(r_0 \text{ codes the pair } (x, z) \text{ and } \forall r_1 \exists r_2(r_2 \notin L[r_1] \wedge r_2 \text{ witnesses that } (1, 1, x, z) \text{ is coded})$ ". Note that the latter formula is of the form  $\exists r_0(\Delta_1^1 \wedge \forall r_1 \exists r_2(\Pi_2^1 \rightarrow \Pi_2^1))$  which is  $\Sigma_5^1$ . Thus the well-order we shall define will become a good  $\Sigma_5^1$ -well-order.

We once more detail our default assumptions for the inductive definition of the iteration. That is we are at stage  $\beta$  of our iteration, let  $G_\beta$  be the  $\mathbb{P}_\beta$ -generic filter and we work over the universe  $W[G_\omega][G_\beta]$ . We also assume that for every  $\eta \leq \beta$  we have defined a notion of  $\infty \cdot \eta$ -allowable forcings avoiding  $(B_\eta \mid \eta < \beta)$  and a countable partial well-order  $<_\beta$  of the reals, i.e. a partial order which is a well-order on its domain. Let the bookkeeping function  $F$  at  $\beta$  hand us  $(\dot{m}, \dot{x})$ . Let  $x$  be  $\dot{x}^{G_\beta}$ ,  $m = \dot{m}^{G_\beta}$  and assume that  $m$  is the Gödel number of a  $\Sigma_5^1$ -formula. In  $W[G_\omega][G_\beta]$  we pick countable partial well-order  $<$  which extends our order  $<_\beta$  and which contains  $x$ .

We will face two cases which we need to discuss. The first case is that  $x$  is an element of the field of  $<_\beta$ . In this case we let  $z$  a code for be the set of  $<_\beta$ -predecessors of  $x$  and force with

$$\dot{Q}_\beta^{G_\beta} := \text{Code}(1, 1, x, z, \eta)$$

for an  $\eta$  which has not been used for coding yet.

In the second case  $x$  will be an element of  $<$  but not of  $<_\beta$ . In this situation we consider the countable set of  $<$ -predecessors of  $x$ , code it with a real  $z$ . We set  $<_{\beta+1} := < \upharpoonright x$ , that is we say that  $<_{\beta+1}$  is just the order  $<$  restricted to elements which are  $\leq x$  and force with

$$\dot{Q}_\beta^{G_\beta} := \text{Code}(1, 1, x, z, \eta)$$

for the least  $\eta$  which is free.

Moreover we ensure that we will not add codes which code a wrong well-order up to  $x$  anymore. That is, whenever  $y$  is a real coding  $\omega$ -many reals which corresponds to an initial segment of a good well-order of the reals,

and  $x' \leq_{\beta+1} x$  appears as one of the elements coded by  $y$ , and the well-order coded by  $y$  below  $x'$  does not coincide with  $<_{\beta+1}$  then we must not use  $\text{Code}(1, 1, x', y, \eta)$  as a forcing. Thus we let

$$B := \{(1, 1, x', \dot{y}) \mid \dot{y} \text{ is the name of a real which codes a well-order below } x' \text{ which is forced to not coincide with } <_{\beta+1}\}$$

and define

$$B_\beta := B.$$

Then define the notion of  $\infty \cdot (\beta + 1)$ -allowable avoiding  $(B_\eta \mid \eta < \beta + 1)$  as the fixed point of the infinite thinning out process with the set of  $\infty \cdot \beta$ -allowable forcings avoiding  $(B_\eta \mid \eta < \beta)$  which additionally avoided the set  $B_\beta$  as the base set.

## 8 Discussion of the universe

In this section we show that the just defined universe has the desired properties. Let  $G_{\omega_1}$  denote a generic filter over  $W[G_\omega]$  for the just defined iteration. We argue in  $W[G_\omega][G_{\omega_1}]$  from now on.

### 8.1 $\Pi_3^1$ -reduction holds in our universe

We first argue why  $\Pi_3^1$ -reduction holds in  $W[G_\omega][G_{\omega_1}]$ .

For every pair  $(m, k) \in \omega^2$ , we define in a first step

$$D_{m,k}^0 := \{x \in \omega^\omega : (m, k, 0, x) \text{ is not coded into the } \vec{S}\text{-sequence}\}$$

and

$$D_{m,k}^1 := \{x \in \omega^\omega : (m, k, 1, x) \text{ is not coded into the } \vec{S}\text{-sequence}\}.$$

Now both sets will not necessarily reduce  $A_m$  and  $A_k$ , as we do have potentially wrong codes that we created when passing from  $W$  to  $W[G_\omega]$  while working towards a failure of  $\Pi_3^1$ -uniformization. There is a  $\Sigma_3^1$ -definition of the real  $R$  that codes all these potentially wrong codes however. This  $\Sigma_3^1$ -definition still works over the bigger universe  $W[G_\omega][G_{\omega_1}]$  and we can use it to define the reducing sets in the following way. For the following we use the phrase “ $(m, k, 0, x)$  is not coded outside of  $L[R]$ ” as an abbreviation of the assertion “ $\neg \exists r (r \text{ witnesses that } (m, k, 0, x) \text{ is coded into } \vec{S} \text{ and } r \notin L[R])$ ”. We define

$$E_{m,k}^0 := \{x \in \omega^\omega : (m, k, 0, x) \text{ is not coded into } \vec{S} \text{ outside of } L[R]\}$$

and

$$E_{m,k}^1 := \{x \in \omega^\omega : (m, k, 1, x) \text{ is not coded into } \vec{S} \text{ outside of } L[R]\}.$$

Note that both  $E_{m,k}^0$  and  $E_{m,k}^1$  are  $\Pi_3^1$ -definable over  $W[G_\omega][G_{\omega_1}]$ . Indeed 7.1.2 shows that

$$\begin{aligned} \forall x(x \in E_{m,k}^0 \Leftrightarrow \forall R(((R)_0, 0, 0) \text{ is coded} \wedge \forall n \in \omega(((R)_{n+1}, n+1, n+1) \\ \text{is coded} \wedge (R)_{n+1} \notin L[(R)_n]) \Rightarrow \neg \exists r(r \notin L[R] \wedge \\ r \text{ witnesses that } (m, k, 0, x) \text{ is coded into } \vec{S}))) \end{aligned}$$

Note that right hand of the above statement is of the form  $\forall R(\Sigma_3^1 \wedge \Sigma_3^1 \wedge \Sigma_3^1 \rightarrow \neg \exists (\Pi_2^1 \wedge \Pi_2^1))$ , hence  $\Pi_3^1$  as desired.

Our goal is to show that for every pair  $(m, k)$  the sets  $E_{m,k}^0 \cap A_m$  and  $E_{m,k}^1 \cap A_k$  reduce the pair of  $\Pi_3^1$ -sets  $A_m$  and  $A_k$ .

**Lemma 8.1.** *In  $W[G_\omega][G_{\omega_1}]$ , for every pair  $(m, k)$ ,  $m, k \in \omega$  and corresponding  $\Pi_3^1$ -sets  $A_m$  and  $A_k$ :*

- (a)  $E_{m,k}^0 \cap A_m$  and  $E_{m,k}^1 \cap A_k$  are disjoint.
- (b)  $(E_{m,k}^0 \cap A_m) \cup (E_{m,k}^1 \cap A_k) = A_m \cup A_k$ .
- (c)  $E_{m,k}^0 \cap A_m$  and  $E_{m,k}^1 \cap A_k$  are  $\Pi_3^1$ -definable.

*Proof.* We prove (a) first. We argue in  $W[G_\omega][G_{\omega_1}]$  but ignore the codes created in  $W[G_\omega]$ . This is justified by the above discussion. If  $x$  is an arbitrary real in  $A_m \cap A_k$  there will be a least stage  $\beta$  above  $\omega$ , such that  $F$  at stage  $\beta$  considers a triple of names which itself corresponds to the triple  $(m, k, x)$ . As  $x \in A_m \cap A_k$ , we know that case 1 or case 2 in the definition of our iteration in the case where we force towards  $\Pi_3^1$ -reduction, must have applied. We argue for case 1 as case 2 is similar. In case 1 the application of  $\text{Code}(m, k, 0, x, \eta)$  codes  $(m, k, x, 0)$  into  $\vec{S}$ , while ensuring that for all future extensions,  $x$  will remain an element of  $A_m$ . The rules of the iteration also tell us that  $(m, k, x, 1)$  will never be coded into  $\vec{S}$  by a later factor of the iteration. Thus  $x \in E_{m,k}^0 \cap A_m$ . It follows that  $x \notin E_{m,k}^1$  and  $E_{m,k}^0 \cap A_m$  and  $E_{m,k}^1 \cap A_k$  are disjoint.

To prove (b), let  $x$  be an arbitrary element of  $A_m \cup A_k$ . Let  $\beta > \omega$  be the stage of the iteration beyond  $W[G_\omega]$  where the triple  $(m, k, x)$  is considered first. As  $x \in A_m \cup A_k$ , either case 1 or case 2 in the the definiton of the forcing for the  $\Pi_3^1$ -reduction were applied at stage  $\gamma$ .

Assume first that it was case 1. Then, as argued above,  $x \in A_m$  will remain true for the rest of the iteration, and we will never code  $(m, k, 1, x)$  into  $\vec{S}$  at a later stage of our iteration. Hence  $x \in A_m \cap D_{m,k}^0$ . If at stage  $\beta$  case 2 applied, then  $x \in E_{m,k}^1 \cap A_k$ , and again, we will never code  $(m, k, 0, x)$

into  $\vec{S}$  at a later stage of our iteration. Thus, either  $x \in E_{m,k}^0 \cap A_m$  or  $x \in E_{m,k}^1 \cap A_k$  and we are finished.

Proving (c) is a straightforward calculation. □

## 8.2 Proof of $\Sigma_4^1$ -uniformization

We claim that adding the two cases to our definition of the iteration will force  $\Sigma_4^1$ -uniformization in the final model.

**Theorem 8.2.** *Let  $G_{\omega_1}$  be a  $\mathbb{P}_{\omega_1}$ -generic filter, where  $\mathbb{P}_{\omega_1}$  should denote the limit of the iteration  $(\mathbb{P}_\alpha, \mathbb{Q}_\beta \mid \beta < \omega_1, \alpha \leq \omega_1)$ . Then in  $W[G_\omega][G_{\omega_1}]$  the  $\Sigma_4^1$ -uniformization property holds, the  $\Pi_3^1$ -reduction property holds and the  $\Pi_3^1$ -uniformization property fails.*

*Proof.* We start arguing for the  $\Sigma_4^1$ -uniformization property. Let  $m$  be the Gödel number of a  $\Sigma_4^1$ -formula in two free variables  $\varphi_m \equiv \exists a_0 \forall a_1 \psi_m(x, y, a_0, a_1)$ . Let  $x$  be an arbitrary real. We assume first that we stay in case 2 throughout the iteration. Then, by definition of the iteration, we force for every  $y'$  and  $a_0$  a real  $a_1$  such that

$$\neg \psi_m(x, y', a_0, a_1)$$

so the  $x$ -section of the set defined by  $\varphi_m$  is empty in  $W[G_\omega][G_{\omega_1}]$ .

If on the other hand we encounter a stage  $\beta$  in our iteration where we are in the first case of the definition, then there must be a least such stage. At this stage we add a real  $a_0$  which witnesses that all further  $\infty \cdot \beta$ -allowable forcings avoiding  $(B_\eta \mid \eta < \beta)$   $\mathbb{Q}'$  will force  $\forall a_1 \psi_m(x, y, a_0, a_1)$ . Then we added a real  $c$  and ensured that eventually

$$\forall a_1 ((m, x, y, c, a_1) \text{ is not coded}).$$

As a consequence

$$\exists a_0 \forall a_1 ((m, x, y, c, a_1) \text{ is not coded}).$$

is true which is a  $\Sigma_4^1$ -property for the reals  $x, y$ . On the other hand, the definition of the iteration ensures that after  $\omega_1$ -many stages, for any real  $y' \neq y = f_m(x)$

$$\forall a_0 \exists a_1 ((m, x, y', c, a_1) \text{ is coded}),$$

thus  $(x, y)$  is the unique pair which satisfies a  $\Sigma_4^1$ -property. As the arguments did not depend on  $m$  or  $x$ , this shows that every  $\Sigma_4^1$ -set in the plane can be uniformized by a function whose graph is a  $\Sigma_4^1$ -set, as desired. □

### 8.3 Proof of the good $\Sigma_5^1$ -wellorder

**Lemma 8.3.** *Let  $W[G_\omega][G_{\omega_1}]$  be the result if we run our definition of the iteration for  $\omega_1$ -many steps over  $W[G_\omega]$ . Then the reals of  $W[G_\omega][G_{\omega_1}]$  have a good  $\Sigma_5^1$ -well-order of the reals. Hence  $\Sigma_m^1$ -uniformization holds for every  $m \geq 5$ .*

*Proof.* In  $W[G_\omega][G_{\omega_1}]$  define

$$< := \bigcup_{\beta < \omega_1} <_\beta .$$

Then  $<$  is a well-order of the reals of ordertype  $\omega_1$  by definition. By construction, if  $x$  is a real and  $z$  a real coding the  $<$ -predecessors of  $x$ , then  $(x, z)$  will satisfy the  $\Sigma_5^1$ -formula “ $(1, 1, x, z)$  is coded cofinally often”.

What is left is to show the converse. That is for any real  $x$ , if  $z$  is a real which does not code a  $<$ -initial segment below  $x$ , then  $(1, 1, x, z)$  is *not* coded cofinally often. This is true, as if  $z$  is such a real then there will be a least stage  $\beta < \omega_1$  such that  $z$  does not coincide with  $<_\beta$  below  $x$ . But from stage  $\beta$  on  $(1, 1, x, z)$  will belong to  $B$ , the set of reals we want to avoid as we put it into  $B_{\beta+1}$  by definition. In particular there is a real  $r$  such that  $(1, 1, x, z)$  is not coded outside of  $L[r]$  in other words, there  $r$  is such that for no real  $s$ ,  $s \notin L[r]$  and  $s$  witnesses that  $(1, 1, x, z)$  is coded into  $\vec{S}$ . Thus  $r$  witnesses that  $(1, 1, x, z)$  is not coded cofinally often as desired.  $\square$

## 9 The second main theorem

In this section we want to sketch a proof of the second main theorem. Its proof can be seen as an easier variant of the proof from the first main theorem. Recall what we are aiming for:

**Theorem 9.1.** *There is a generic extension of  $L$  where the  $\Pi_3^1$ -uniformization property holds and where  $\Sigma_n^1$ -uniformization holds for  $n \geq 4$ .*

The theorem is proved using an  $\omega_1$ -length iteration of allowable forcings; the notion of allowable needs to be altered though for our new task. It should accommodate the following tasks:

1. Forcing  $\Pi_3^1$ -uniformization
2. Forcing a good  $\Sigma_5^1$  well-order of the reals.

Hence we define a new variant of 0-allowable forcings designed to carry out such a proof.

**Definition 9.2.** Let  $W$  be our ground model. Let  $\alpha < \omega_1$  and let  $F \in W$ ,  $F : \alpha \rightarrow W$  be a bookkeeping function. A finite support iteration  $\mathbb{P} = (\mathbb{P}_\beta : \beta < \alpha)$  is called allowable (relative to the bookkeeping function  $F$ ) if the function  $F : \alpha \rightarrow W$  determines  $\mathbb{P}$  inductively as follows:

- (1) We assume that  $\beta \geq 0$  and  $\mathbb{P}_\beta$  is defined. We let  $G_\beta$  be a  $\mathbb{P}_\beta$ -generic filter over  $W$  and assume that  $F(\beta) = (\dot{m}, \dot{x}, \dot{y}, \dot{\eta})$ , for a quadruple of  $\mathbb{P}_\beta$ -names. We assume that  $\dot{x}^{G_\beta} =: x$ ,  $\dot{y}^{G_\beta} = y$  are reals,  $\dot{m}^{G_\beta} =: m$  is a natural number which codes a  $\Pi_3^1$ -set  $A_m$  and  $\dot{\eta}^{G_\beta}$  is an ordinal  $< \omega_1$ .

Then we split into two cases:

- If there is a  $\gamma < \beta$  and a  $\mathbb{P}_\gamma$ -name of a triple  $(\dot{m}', \dot{a}, \dot{b}, \dot{\eta})$  such that  $\dot{a}^{G_\gamma} = a \in \omega^\omega$ ,  $\dot{b}^{G_\gamma} = b$ ,  $\dot{m}'^{G_\gamma} = m' \in \omega$ ,  $\dot{\eta}^{G_\gamma} = \eta$  and  $F(\gamma) = (\dot{m}, \dot{a}, \dot{b}, \dot{\eta})$ , then we force with the trivial forcing. We say in this situation that  $\eta$  has already been used for coding, or  $\eta$  is not free.
- If not, then let

$$\dot{\mathbb{Q}}_\beta^{G_\beta} = \mathbb{P}(\beta)^{G_\beta} := \text{Code}(m, x, y, \eta).$$

We say that in this situation  $\eta$  is free or  $\eta$  has not been used for coding yet.

- (2) If  $F(\beta) = (\dot{m}, \dot{x}, \dot{y}, \dot{\eta})$  and  $\dot{x}, \dot{y}$  are both names of reals whereas  $\dot{m}$  is the Gödel number of a  $\Sigma_5^1$ -formula, then we use  $\text{Code}(1, 1, x, y, \eta)$ , provided  $\eta$  is free and use the least  $\eta'$  which is free otherwise for coding  $(1, x, y)$  there. Again to avoid ambiguity we assume that 1 is not the Gödel number of a formula and code  $(1, 1, x, y)$  instead of just  $(1, x, y)$ .

This version of “allowable” is less complicated than the version we needed for the proof of the first theorem. The reason is that we just have two tasks we need to take care of, and that the most tricky case, namely forcing reduction and a failure of uniformization is not existing.

We proceed now in a very similar way to the proof of the first main theorem, but the thinning out process is tailored to finally obtain a universe where the  $\Pi_3^1$ -uniformization property is true. We employ the thinning out process as detailed in [9] or [10]. Then define a version of “thinning out while leaving out codes” which is a straightforward adaption of the construction present in this paper.

Equipped with these notions, we start to prove the second main theorem following closely the proof of the first one. As we only have to deal with the  $\Pi_3^1$ -uniformization and a  $\Sigma_5^1$ -good wellorder, the proof is shorter.

## 9.1 Thinning out for uniformization

We define the derivative acting on the set of allowable forcings over  $W$ . Inductively, for an ordinal  $\alpha$  and any bookkeeping function  $F \in W$ , we assume that the notion of  $\zeta$ -allowable with respect to  $F$  has already been defined for every  $\zeta < \alpha$ . Specifically, this means that for each  $\zeta < \alpha$ , we have already defined a set of rules that, in conjunction with a bookkeeping function  $F \in W$ , produces the following over  $W$ :

- An allowable forcing  $\mathbb{P} = \mathbb{P}_\delta = ((\mathbb{P}_\beta, \dot{Q})_\beta : \beta < \delta) \in W$ , which is the actual forcing used in the iteration. Let  $G_\delta$  denote a  $\mathbb{P}_\delta$ -generic filter over  $W$ .
- A set

$$I = \dot{I}_\delta^{G_\delta} = \{(\dot{x}^{G_\delta}, \dot{y}^{G_\delta}, \dot{m}^{G_\delta}, \dot{\gamma}^{G_\delta}) : \dot{m}, \dot{x}, \dot{y}, \dot{\gamma} \text{ are } \mathbb{P}\text{-names for elements of } \omega, 2^\omega, \omega_1\}.$$

The set  $I \in W[G_\delta]$  contains potential values for the uniformizing function  $f$  that we want to define. Note that for a given  $x$  and  $m$ , there may be several values  $(x, y_1, m, \xi_1), \dots, (x, y_n, m, \xi_n)$ . We say that  $(x, y, m)$  has rank  $\xi$  if  $(x, y, m, \xi) \in I$ . There can be multiple ranks for a given  $(x, y, m)$ . The goal is to use the  $(x, y, m)$ 's with the minimal rank, and among those, choose the one with the least name according to the fixed well-order of  $L$  in the background, ensuring the well-definedness of our choice.

Following our established terminology, if applying the rules for  $\eta$ -allowable forcings over  $W$  and  $F \in W$  results in the pair  $(\mathbb{P}, I) \in W$ , we say that  $\mathbb{P}$  is  $\eta$ -allowable with respect to  $F$  (over  $W$ ), or simply that  $\mathbb{P}$  is  $\eta$ -allowable if there exists an  $F$  and an  $I$  such that  $\mathbb{P}$  is  $\eta$ -allowable with respect to  $F$ .

We now define the derivation of the  $< \alpha$ -allowable forcings over  $W$ , which we call  $\alpha$ -allowable (again over  $W$ ). The definition is a uniform extension of 0-allowability.

A  $\delta < \omega_1$ -length iteration  $\mathbb{P} = (\mathbb{P}_\beta : \beta < \delta) \in W$  is called  $\alpha$ -allowable over  $W$  (or relative to  $W$ ) if it is recursively constructed using two ingredients. First, a bookkeeping function  $F \in W$ ,  $F : \delta \rightarrow W^3$ , where for each  $\beta < \delta$ , we write  $F(\beta) = ((F(\beta)_0, F(\beta)_1, F(\beta)_2))$  for the corresponding values of the coordinates. Second, a set of rules similar to those for 0-allowability, with two additional rules added at each step of the derivative, which determine, along with  $F$ , how the iteration  $\mathbb{P}$  and the set of  $f$ -values  $I$  are constructed.

The infinite set of rules is defined as follows. Fix a bookkeeping function  $F \in W$ ,  $F : \delta \rightarrow W^3$ , for  $\delta < \omega_1$ . Assume we are at stage  $\beta$  of our construction and that, inductively, we have already created the following list of objects:

- The forcing  $\mathbb{P}_\beta \in W$  up to stage  $\beta$ , along with a  $\mathbb{P}_\beta$ -generic filter  $G_\beta$  over  $W$ .
- The set

$$I = \dot{I}_\delta^{G_\delta} = \{(\dot{x}^{G_\delta}, \dot{y}^{G_\delta}, \dot{m}^{G_\delta}, \dot{\gamma}^{G_\delta}) : \dot{m}, \dot{x}, \dot{y}, \dot{\gamma} \text{ are } \mathbb{P}\text{-names for elements of } \omega, 2^\omega, \omega_1\},$$

containing potential values for the uniformizing function  $\dot{f}^{G_\beta}(m, \cdot)$ . Initially, we set  $I_0 = \emptyset$ .

The set of possible  $f$ -values will change as the iteration progresses. Specifically, values for  $f$  must be added when a new, lower-ranked value of  $\dot{f}^{G_\beta}(m, x)$  is encountered.

Now, working in  $W[G_\beta]$ , we define the next forcing  $\dot{\mathbb{Q}}^{G_\beta}$  and possibly update the set of possible values for the uniformizing function  $f(m, x)$ . Assume that  $F(\beta)_0 = (\dot{x}, \dot{y}, \dot{m})$ , and let  $A \subset \beta$ ,  $A \in W$ , be such that  $\dot{x}, \dot{y}, \dot{m}$  are  $\mathbb{P}_A = *_{\eta \in A} \mathbb{P}(\eta)$ -names, where we require that  $\mathbb{P}_A \in W$  is a subforcing of  $\mathbb{P}_\beta$  (e.g., if  $A = \gamma < \beta$  and  $F(\beta)_0$  lists  $\mathbb{P}_\gamma$ -names). Let  $G_A := G_\beta \upharpoonright A$ . We then set  $x = \dot{x}^{G_A}$ ,  $y = \dot{y}^{G_A}$ , and  $m = \dot{m}^{G_A}$ , and proceed as follows:

- (a) There exists an ordinal  $\zeta < \alpha + 1$ , chosen to be minimal, such that:

First, we collect all  $\mathbb{P}_A$ -names for reals  $\dot{a}$ . For each  $\mathbb{P}_A$ -name  $\dot{a}$ , we pick the  $<$ -least nice name  $\dot{b}$  such that  $\dot{a}^{G_\beta} = \dot{b}^{G_\beta}$ , and collect these names  $\dot{b}$  into a set  $C$ . We assume that there is a  $<$ -least nice  $\mathbb{P}_A$ -name  $\dot{y}_0$  in  $C$  such that  $\dot{y}_0^{G_A} = y_0$ ,

$$W[G_\beta] \models (x, y_0) \in A_m$$

and there is no  $\zeta$ -allowable forcing  $\mathbb{R}$ ,  $\mathbb{R} \in W$ , extending  $\mathbb{P}_\beta$  such that

$$W[G_\beta] \models \mathbb{R}/G_\beta \Vdash (x, y_0) \notin A_m.$$

If this condition holds, we proceed as follows:

- Assume that  $F(\beta)_2 = (\dot{x}, \dot{z}, \dot{m})$  is a triple of  $\mathbb{P}_A$ -names, with  $\dot{x}^{G_A} = x$ ,  $\dot{z}^{G_A} = z \neq y_0$ , and  $\dot{m}^{G_A} = m$ . We define:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, x, z, \eta).$$

If the bookkeeping function does not have the desired form, we choose the  $<$ -least names of the desired objects and use them to define the forcing. In this case, we pick the  $<$ -

least  $\mathbb{P}_A$ -name for a countable ordinal  $\dot{\eta}$ , and let  $\dot{z}$  be the  $<$ -least  $\mathbb{P}_A$ -name of a real such that  $\dot{z}^{G_A} \neq y_0$ . Then:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, x, z, \eta).$$

We also set  $\mathbb{P}_{\beta+1} = \mathbb{P}_\beta * \dot{\mathbb{Q}}_\beta$  and let  $G_{\beta+1} = G_\beta * G(\beta)$  be its generic filter.

- We assign a new value to  $f$ , i.e., set  $f(m, x) := y_0$  and assign the rank  $\zeta$  to the value  $(x, y_0, m)$  in  $W[G_{\beta+1}]$ . We update  $I_{\beta+1}^{G_{\beta+1}} := I_\beta^{G_\beta} \cup \{(x, y_0, m, \zeta)\}$ .

- (b) If case (a) does not apply, i.e., for each  $\zeta < \alpha$  and each pair of reals, the pair can be forced out of  $A_m$  by a  $\zeta$ -allowable forcing extending the current one, we let the bookkeeping function  $F$  fully determine what to force. We assume that  $F(\beta)_1$  is a  $\mathbb{P}_A$ -name for a countable ordinal  $\dot{\eta}$ , and let  $\dot{\eta}^{G_A} = \eta$ . We assume that  $F(\beta)_2$  is a nice  $\mathbb{P}_A$ -name for a pair of reals  $(\dot{x}', y_0)$  such that  $\dot{x}'^{G_A} = x$ . We define:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, x, y_0, \eta).$$

Let  $G(\beta)$  be a  $\dot{\mathbb{Q}}_\beta$ -generic filter over  $W[G_\beta]$  and set  $G_{\beta+1} = G_\beta * G(\beta)$ . We do not update the set  $I_\beta$  of preliminary values for  $f$ , i.e., we set  $I_{\beta+1} := I_\beta$ .

Otherwise, we choose the  $<$ -least  $\mathbb{P}_A$ -names for the desired objects  $g_\eta$  and  $(x, z, m)$ , and force with:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, x, z, \eta).$$

- (c) If  $F(\beta) = (\dot{m}, \dot{x}, \dot{y}, \dot{\eta})$  for  $\dot{m}$  a  $\mathbb{P}_\beta$ -name of an integer,  $\dot{x}, \dot{y}$  tow names of reals and  $\dot{\eta}$  the name of a countable ordinals then we force with:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(1, 1, x, y, \eta), \text{ provided } \eta \text{ is free}$$

$$\text{or } \dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(1, 1, x, y, \eta'), \text{ for } \eta' \text{ least that is free for coding.}$$

At limit stages  $\eta$  of  $\alpha + 1$ -allowable forcings, we use finite support:

$$\mathbb{P}_\eta := \text{dir lim}(\mathbb{P}_\nu : \nu < \eta).$$

Finally, we set:

$$I_\eta^{G_\eta} := \{(m, x, y, \zeta) : \exists \xi < \eta ((m, x, y, \zeta) \in I_\xi^{G_\xi})\}.$$

This concludes the definition of the rules for  $\alpha + 1$ -allowability over the ground model  $W$ . To summarize:

**Definition 9.3.** Assume that  $F \in W$ ,  $F : \eta \rightarrow W^3$  is a bookkeeping function and that  $\mathbb{P} = (\mathbb{P}_\beta : \beta < \eta)$  and  $I = (I_\beta : \beta < \eta)$  is the result of applying the above defined rules together with  $F$  over  $W$ . Then we say that  $(\mathbb{P}, I)$  is  $\alpha + 1$ -allowable with respect to  $F$  (over  $W$ ). Often,  $I$  is clear from context, and we will just say  $\mathbb{P}$  is  $\alpha + 1$ -allowable with respect to  $F$ . We also say that  $\mathbb{P}$  is  $\alpha + 1$ -allowable over  $W$  if there is an  $F$  such that  $\mathbb{P}$  is  $\alpha + 1$ -allowable with respect to  $F$ .

## 9.2 Thinning out while leaving out codes for uniformization

As in the proof of the first main theorem, we shall define the notion of an allowable forcing which avoids a set  $B$  of names of reals.

**Definition 9.4.** Let  $B$  be an arbitrary set of tuples of (forcing names of) reals denoted by  $\vec{x}$ . We say that an iteration  $\{(\mathbb{P}_\alpha, \dot{Q}_\alpha) \mid \alpha < \delta\}$  is 0-allowable without using  $B$  (or avoiding  $B$ ) if it is allowable and for every  $\alpha < \delta$  and every  $\mathbb{P}_\alpha$ -generic filter  $G_\alpha$ , none of the factors  $\dot{Q}_\alpha^{G_\alpha}$  are of the form  $\text{Code}(1, 1, \vec{x}^{G_\alpha}, \eta)$ ,  $\eta < \omega_1$  and  $\vec{x} \in B$ , where we write  $\vec{x}^{G_\alpha}$  for the evaluation of the  $\mathbb{P}_\alpha$ -names which are elements of  $\vec{x}$  with the help of the generic  $G_\alpha$ .

## 9.3 Proof of the second main theorem

We first assume that  $F(\beta) = (\beta_1, \beta_2)$  and that the  $\beta_1$ -th (in some previously fixed well-order  $<$  of  $H(\omega_2)$ )  $\mathbb{P}_{\beta_0}$ -name of a triple of the form  $(\dot{n}, \dot{l}, \dot{a})$ , where  $\dot{a}$  is a nice  $\mathbb{P}_{\beta_0}$ -name of a real, and  $\dot{n}, \dot{l}$  are nice  $\mathbb{P}_{\beta_0}$ -names of natural numbers, is the triple  $(\dot{m}, \dot{k}, \dot{x})$ . We let  $m = \dot{m}^{G_\beta}, k = \dot{k}^{G_\beta}$  and assume that  $m, k$  are both Gödel numbers of  $\Pi_3^1$ -formulas and  $x = \dot{x}^{G_\beta}$  is a real. In this situation we work towards  $\Pi_3^1$ -uniformization. We work over  $W[G_\beta]$  as the ground model. This model comes along with a sequence of sets of tuples of (names of) reals

$$B := (B_{\zeta_0} \mid \zeta_0 < \beta)$$

and the notion of  $\infty \cdot \zeta_0 + \zeta_1$ -allowable avoiding  $(B_i \mid i < \zeta_0)$  for  $\zeta_0 < \beta, \zeta_1 \in \text{Ord}$ . We distinguish several according to the thinning out process for uniformization.

- (a) There exist two ordinals  $\zeta_0 < \beta, \zeta_1 \in \text{Ord}$  chosen to be lexicographically minimal such that:

First, we collect all  $\mathbb{P}_A$ -names for reals  $\dot{a}$ . For each  $\mathbb{P}_A$ -name  $\dot{a}$ , we pick the  $<$ -least nice name  $\dot{b}$  such that  $\dot{a}^{G_\beta} = \dot{b}^{G_\beta}$ , and collect these names  $\dot{b}$  into a set  $C$ . We assume that there is a  $<$ -least nice  $\mathbb{P}_A$ -name  $\dot{y}_0$  in  $C$  such that  $\dot{y}_0^{G_A} = y_0$ ,

$$W[G_\beta] \models (x, y_0) \in A_m$$

and there is no  $\infty \cdot \zeta_0 + \zeta_1$  allowable forcing avoiding  $B$  denoted by  $\mathbb{R}$ , extending  $\mathbb{P}_\beta$  such that

$$W[G_\beta] \models \mathbb{R}/G_\beta \Vdash (x, y_0) \notin A_m.$$

If this condition holds, we proceed as follows:

- Assume that  $F(\beta)_2 = (\dot{x}, \dot{z}, \dot{m})$  is a triple of  $\mathbb{P}_A$ -names, with  $\dot{x}^{G_A} = x$ ,  $\dot{z}^{G_A} = z \neq y_0$ , and  $\dot{m}^{G_A} = m$ . We define:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, x, z, \eta).$$

If the value of bookkeeping function at  $\beta$  does not have the desired form, we choose the  $<$ -least names of the desired objects and use them to define the forcing. In this case, we pick the  $<$ -least  $\mathbb{P}_A$ -name for a countable ordinal  $\dot{\eta}$ , and let  $\dot{z}$  be the  $<$ -least  $\mathbb{P}_A$ -name of a real such that  $\dot{z}^{G_A} \neq y_0$ . Then:

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(m, x, z, \eta).$$

We also set  $\mathbb{P}_{\beta+1} = \mathbb{P}_\beta * \dot{\mathbb{Q}}_\beta$  and let  $G_{\beta+1} = G_\beta * G(\beta)$  be its generic filter.

- We assign a new value to  $f$ , i.e., set  $f(m, x) := y_0$  and assign the rank  $\zeta$  to the value  $(x, y_0, m)$  in  $W[G_{\beta+1}]$ . We update  $I_{\beta+1}^{G_{\beta+1}} := I_\beta^{G_\beta} \cup \{(x, y_0, m, \zeta)\}$ .

- (b) If case (a) does not apply, i.e., for each  $\zeta_0 < \beta, \zeta_1 \in \text{Ord}$  and each pair of reals  $(x, y) \in W[G_\beta]$ , the pair can be forced out of  $A_m$  by a  $\infty \cdot \zeta_0 + \zeta_1$ -allowable forcing  $\mathbb{P}_{\zeta_0, \zeta_1}$  extending the current one. Then we decide to pick one such pair  $(x, y)$  using the bookkeeping  $F$  and fix the  $<$ -least such forcing  $\mathbb{P}_{\infty, \beta}(x, y)$ , which is  $\infty \cdot \beta$ -allowable avoiding  $B_\beta$  and let  $C^\omega(\mathbb{P})(x, y)$  denote, as before, the closure of  $\mathbb{P}$ , which is defined in the analogous way to the notion of closure we defined for the reduction property. Then, at stage  $\beta$ , we force with

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := C^\omega(\mathbb{P}_{\infty, \beta}(x, y)).$$

We note that this forcing forces  $(x, y)$  out of  $A_m$ .

We define the set  $I_{\beta+1}$  in the obvious way. We do not add a new set  $B_\beta$  we need to avoid, that is we let  $B_\beta = \emptyset$ , define the new  $B := (B_\zeta \mid \zeta < \beta + 1)$ , and keep the notions of  $\infty \cdot \zeta_0 + \zeta_1$ -allowable avoiding  $(B_\eta \mid \eta < \zeta_0)$  for each  $\zeta_0 < \beta + 1, \zeta_1 \in \text{Ord}$ .

- (c) We assume here that  $F(\beta) = (\dot{m}, \dot{x})$ . Let  $x$  be  $\dot{x}^{G_\beta}$ ,  $m = \dot{m}^{G_\beta}$  and assume that  $m$  is the Gödel number of a  $\Sigma_5^1$ -formula. In  $W[G_\omega][G_\beta]$  we pick countable partial well-order  $<$  which extends our order  $<_\beta$  and which contains  $x$ .

We will face two cases which we need to discuss. The first case is that  $x$  is an element of the field of  $<_\beta$ . In this case we let  $z$  a code for be the set of  $<_\beta$ -predecessors of  $x$  and force with

$$\dot{Q}_\beta^{G_\beta} := \text{Code}(1, 1, x, z, \eta)$$

for an  $\eta$  which has not been used for coding yet.

In the second case  $x$  will be an element of  $<$  but not of  $<_\beta$ . In this situation we consider the countable set of  $<$ -predecessors of  $x$ , code it with a real  $z$ . We set  $<_{\beta+1} := < \upharpoonright x$ , that is we say that  $<_{\beta+1}$  is just the order  $<$  restricted to elements which are  $\leq x$  and force with

$$\dot{Q}_\beta^{G_\beta} := \text{Code}(1, 1, x, z, \eta)$$

for the least  $\eta$  which is free.

Moreover we ensure that we will not add codes which code a wrong well-order up to  $x$  anymore. That is, whenever  $y$  is a real coding  $\omega$ -many reals which corresponds to an initial segment of a good well-order of the reals, and  $x' \leq_{\beta+1} x$  appears as one of the elements coded by  $y$ , and the well-order coded by  $y$  below  $x'$  does not coincide with  $<_{\beta+1}$  then we must not use  $\text{Code}(1, 1, x', y, \eta)$  as a forcing. Thus we let

$$B := \{(1, 1, x', \dot{y}) \mid \dot{y} \text{ is the name of a real which codes a well-order below } x' \text{ which is forced to not coincide with } <_{\beta+1}\}$$

and define

$$B_{\beta+1} := B_\beta \cup B.$$

At limit stages  $\eta$  of  $\alpha + 1$ -allowable forcings, we use finite support:

$$\mathbb{P}_\eta := \text{dir lim}(\mathbb{P}_\nu : \nu < \eta).$$

Finally, we set:

$$I_\eta^{G_\eta} := \{(m, x, y, \zeta) : \exists \xi < \eta ((m, x, y, \zeta) \in I_\xi^{G_\xi})\}.$$

## 9.4 Proof of the second main theorem

We iterate the just defined forcing  $\omega_1$ -many times using a bookkeeping  $F : \omega_1 \rightarrow H(\omega_1)$  which satisfies that every  $x \in H(\omega_1)$  has uncountable pre-image under  $F$ . This assumption ensures that we consider every tuple of reals cofinally often. The resulting partial order will be denoted by  $\mathbb{P}_{\omega_1}$  and is the desired forcing.

**Theorem 9.5.** *Let  $G_{\omega_1}$  be a  $\mathbb{P}_{\omega_1}$ -generic filter over  $W$ . In  $W[G_{\omega_1}]$   $\Pi_3^1$ -uniformization holds, and there is a  $\Sigma_5^1$ -definable good wellorder of the reals.*

*Proof.* We shall sketch the proof of  $\Pi_3^1$ -uniformization first, which is basically the same as in [9] or [10] and the reader can find all the details there. For  $m$  the Gödel number of an arbitrary  $\Pi_3^1$ -set in the plane, we define its uniformization

$$f_m(x) = y \Leftrightarrow \neg \exists r (\forall M (\text{If } M \text{ is countable and transitive and} \\ M \models \text{ZF}^- + \text{“}\aleph_1 \text{ exists” and } \omega_1^M = (\omega_1^L)^M \text{ and} \\ r, (m, x, y) \in M) \\ \text{then} \\ M \models \text{“}L[r] \text{ decodes out of } r \text{ an } \aleph_1\text{-sized, transitive} \\ \text{ZF}^- \text{-model } N \text{ which witnesses that} \\ (m, x, y) \text{ got coded into } \vec{S}\text{”}).$$

By definition, in our iteration, for every real  $x$ , there will be at most one  $y$  which satisfies  $f_m(x) = y$ , and this only happens if the  $x$ -section of  $A_m$  is non-empty in  $W[G_{\omega_1}]$ .

The proof that  $W[G_{\omega_1}]$  can define a good  $\Sigma_5^1$ -wellorder of the reals is exactly as in the first main theorem.  $\square$

## 10 Lifting to $M_n$

We just state here that the proofs of the two main theorems are such that they can be lifted with some care to canonical inner models with Woodin cardinals. The interested reader can consult [9] for a detailed proof of how one can lift the proof of  $\Pi_3^1$ -uniformization. The adaptations one has to undertake in order to lift the proofs of the two results above are rather straightforward then.

**Theorem 10.1.** *Let  $M_n$  denote the canonical inner model with  $n$ -many Woodin cardinals. There is a set-generic extension  $M_n[g]$  of  $M_n$  which preserves the Woodin cardinals and where additionally*

1.  $\Pi_{n+3}^1$ -reduction holds,

2.  $\Pi_{n+3}^1$ -uniformization fails,
3.  $\Sigma_{m+4}^1$ -uniformization holds for every  $m \geq n$ .

and

**Theorem 10.2.** *There is a set-generic extension  $M_n[g]$  of  $M_n$  preserving the Woodin cardinals and where additionally*

1.  $\Pi_{n+3}^1$ -uniformization holds,
2.  $\Sigma_{m+4}^1$ -uniformization holds for every  $m \geq n$ .

## 11 Open questions

We end with several questions which are related to this article. The first observation is concerned with a hypothetical forcing  $\mathbb{P}$  which would force the  $\Pi_5^1$ -uniformization property over  $L$ . The construction of such a forcing  $\mathbb{P}$  necessarily cannot be combined with the methods presented here, as  $\Sigma_5^1$  and  $\Pi_5^1$ -uniformization contradict each other. Given the flexibility of the way the good  $\Sigma_5^1$ -wellorder is forced here, which can even be arranged such that one can force towards  $\Pi_3^1$ -uniformization, it is not clear at all how such a hypothetical forcing would look like. The present work seems to suggest that, if it can be done at all, a very new approach is in need.

**Question 1.** *Can one force  $\Pi_5^1$ -uniformization over just  $L$ ?*

A second question concerns patterns of the  $\Sigma$ -uniformization property.

**Question 2.** *Given a real  $r \in 2^\omega$ , can one force a universe where  $\Sigma_n^1$ -uniformization is true whenever  $r(n) = 1$  and  $\Sigma_n^1$ -uniformization fails whenever  $r(n) = 0$ ?*

By a classical result of Novikov, for any projective pointclass  $\Gamma$ , it is impossible to have  $\Gamma$  and  $\check{\Gamma}$ -reduction simultaneously. The case for separation is still unknown.

**Question 3.** *Can one force a universe where  $\Sigma_3^1$ - and  $\Pi_3^1$ -separation hold simultaneously?*

## 12 Acknowledgment

To the one whose very short life prompted this work. Your family loves you. This research was funded in whole by the Austrian Science Fund (FWF) Grant-DOI 10.55776/P37228.

## References

- [1] J. Addison *Some consequences of the axiom of constructibility*, Fundamenta Mathematica, vol. 46 (1959), pp. 337–357.
- [2] R. David *A very absolute  $\Pi_2^1$ -real singleton*. Annals of Mathematical Logic 23, pp. 101-120, 1982.
- [3] S. Hoeffelner  *$NS_{\omega_1}$   $\Delta_1$ -definable and saturated*. Journal of Symbolic Logic 86 (1), pp. 25 - 59, 2021.
- [4] S. Hoeffelner *Forcing the  $\Sigma_3^1$ -separation property*. Journal of Mathematical Logic 22, No. 2, 2022.
- [5] S. Hoeffelner *Forcing the  $\Pi_3^1$ -Reduction Property and a Failure of  $\Pi_3^1$ -Uniformization*, Annals of Pure and Applied Logic 174 (8), 2023.
- [6] S. Hoeffelner *Forcing Axioms and the Uniformization Property* Annals of Pure and Applied Logic 175 (10), 2024.
- [7] S. Hoeffelner *The global  $\Sigma_{n+2}^1$ -Uniformization Property and BPFA* Advances in Mathematics 470, 2025.
- [8] S. Hoeffelner  *$MA(\mathcal{I})$  and a Failure of Separation on the third Level* . Annals of Pure and Applied Logic 176 (3), 2026.
- [9] S. Hoeffelner *Forcing the  $\Pi_n^1$ -Uniformization Property*, Arxiv preprint. DOI: <https://doi.org/10.48550/arXiv.2103.11748>
- [10] S. Hoeffelner *A Universe with a  $\Delta_n^1$ -definable Well-order of the Reals, CH and  $\Pi_n^1$ -Uniformization*. Arxiv preprint. DOI: <https://doi.org/10.48550/arXiv.2506.21778>
- [11] S. Hoeffelner *A Failure of  $\Pi_{n+3}^1$ -Reduction in the Presence of  $\Sigma_{n+3}^1$ -Separation*, Arxiv preprint. DOI: <https://doi.org/10.48550/arXiv.2312.02540>
- [12] T. Jech *Set Theory. Third Millenium Edition*. Springer 2006.
- [13] R. Jensen and R. Solovay *Some Applications of Almost Disjoint Sets*. Studies in Logic and the Foundations of Mathematics Volume 59, pp. 84-104, 1970.
- [14] A. Kechris *Classical Descriptive Set Theory*. Springer 1995.
- [15] D. Martin and J. Steel *A Proof of Projective Determinacy*. Journal of the American Mathematical Society (2), pp.71-125, 1989.
- [16] Y. Moschovakis *Descriptive Set Theory*. Mathematical Surveys and Monographs 155, AMS.

- [17] Y. Moschovakis *Uniformization in a playful Universe*. Bulletin of the American Mathematical Society 77, no. 5, 731-736, 1971.
- [18] R. Schindler *Set Theory: Exploring Independence and Truth*. Springer 2014.